



TAMPEREEN
AMMATTIKORKEAKOULU

KETTERÄ PROJEKTIHALLINTA HÄMEEN AMMATTIKORKEAKOULUSSA

Kokemuksia WWW-palvelujen
uudistamisesta

Mika Rintala

Opinnäytetyö
Marraskuu 2015

Tietojärjestelmäosaamisen koulutus, ylempi AMK



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojärjestelmäosaamisen koulutus, ylempi AMK

RINTALA MIKA:

Ketterä projektinhallinta Hämeen ammattikorkeakoulussa
Kokemuksia WWW-palvelujen uudistamisesta

Opinnäytetyö 58 sivua, joista liitteitä 5 sivua
Marraskuu 2015

Julkishallinnollisten organisaatioiden on tultava toimeen entistä vähemmillä resursseilla. Tämä koskee myös tietojärjestelmien ylläpitoa ja kehittämistä. Organisaatioiden toimintaympäristö muuttuu yhä kiihtyvällä vauhdilla, eikä tälle muutokselle ole näkyvissä loppua. Perinteiset projektinhallinnan ja -kehittämisen menetelmät soveltuvat huonosti alati muuttuvan maailman tarpeita vastaavien palveluiden kehittämiseen. Näihin haasteisiin vastatakseen Hämeen ammattikorkeakoulu on päättänyt nykyaikaistaa tietojärjestelmäprojektiansa projektinhallintaa sekä tietojärjestelmien kehittämisen menetelmiään. Juuri siihen päätökseen liittyy osaltaan tämä opinnäytetyö.

Uusia menetelmiä pilotoitiin verkkopalveluiden uudistamisprojektissa, joka alkoi vuoden 2012 syksyllä ja jonka viimeinen osaprojekti päättyi joulukuussa 2015. Tämän opinnäytetyön laatija oli mukana projektin ensimmäisessä vaiheessa, jossa uusittiin julkisten verkkopalveluiden tekninen alusta, ulkoasu ja sisältö. Projektin hallintaan käytettiin sovellettua Scrum-menetelmää, ja asiantuntijapalveluiden hankinnat tehtiin aiemmasta poikkeavalla tavalla, kilpailuttamalla vain tarvittavan osaamisen hinta ja mahdollisten puitesopimuskumppaneiden osaamisen taso eli laatu.

Ketterät menetelmät osoittautuivat käyttökelpoisiksi varsinkin tällaisessa projektissa, jossa tarkkaa lopputulosta ei aloitettaessa tiedetä vaan selvillä on vain visio lopputuloksesta. Ketterien menetelmien käyttöä ei kuitenkaan voi varauksetta suositella – näiden menetelmien käyttöönotto vaatii organisaatiolta halua muuttaa toimintatapojaan.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Master's Degree Programme in Information System Competence

RINTALA MIKA:
Agile Project Management in Häme University of Applied Sciences

Bachelor's thesis 58 pages, appendices 5 pages
November 2015

Nowadays IT departments need to cope with fewer resources, and with these scarce resources they need to produce better services than before. That is why Häme University of Applied Sciences (HAMK) decided to evaluate new methodologies for IT project management, development and procurement.

These new methods were piloted within a project, the goal of which was to modernize HAMK's public facing website and intranet. Scrum methodology was chosen to be used in this project. New and more agile methods for procurement were also developed and used.

A case study approach was utilised in this thesis. The Scrum method proved to work better than more traditional methods of project management and development. The new procurement method was a key factor in this project's success and it will most likely be utilized in future projects – especially in those which do not have a more refined goal than a vision when the project starts.

Key words: agile methodology, project management, procurement

SISÄLLYS

1	JOHDANTO.....	8
2	PROJEKTIHALLINTAMENETELMIEN LÄHTÖTILANNE HÄMEEN AMMATTIKORKEAKOULUN TIETOHALLINNOSSA.....	10
2.1	Budjettiseuranta	10
2.2	Riskienhallinta	11
2.3	Aikataulun seuranta	12
2.4	Projektin töiden koordinointi	12
3	KETTERÄT PROJEKTIHALLINTA- JA KEHITTÄMISMENETELMÄT	14
3.1	Ketterät projektinhallintamenetelmät.....	14
3.2	Scrum-menetelmän esittely.....	16
3.2.1	Scrumin vaiheet.....	16
3.2.2	Sprintin suunnittelu ja työskentely sprintin aikana	18
3.2.3	Sprint review	19
3.2.4	Sprint retrospective	19
3.2.5	Henkilöiden roolit scrum-menetelmässä.....	20
3.2.6	Tuoteomistaja	21
3.2.7	Scrum master.....	22
3.2.8	Toteutustiimi	22
3.3	Ketterät menetelmät verrattuna vesiputousmalliin	24
3.4	Ketterien menetelmien keskeisimmät hyödyt ja haitat	26
4	CASE VERKKOPALVELUIDEN UUDISTAMISPROJEKTI	28
4.1	Ketterät menetelmät tässä projektissa.....	29
4.2	Ketterää projektia tukeva hankintamalli	30
4.2.1	Puitesopimuskilpailutus	31
4.2.2	Minikilpailutukset	32
4.2.3	Kokemukset ketteristä hankinnoista	33
4.2.4	Ketterien ja perinteisten hankintojen olennaisimmat erot.....	34
4.2.5	Ketterät hankinnat tulevaisuudessa	35
4.3	Kokemukset ketteristä menetelmistä	36
4.3.1	Toteutustiimin kokemukset.....	37
4.3.2	Toteutustiimin kehittyminen ketterien menetelmien hyödyntäjänä.....	39
4.3.3	Omat kokemukseni ja mitä tekisin toisin	40
5	TOIMIVIA KÄYTÄNTEITÄ HÄMEEN AMMATTIKORKEAKOULULLE	43
5.1	Ketterä kilpailuttaminen	43

5.2	Projektien priorisointi	44
5.3	Tuoteomistajan ja tiimin voimaannuttaminen	44
5.4	Kehitystiimin työn organisoiminen	45
5.5	Tiimin ja projektijohdon kehittyminen	47
6	POHDINTA.....	50
	LÄHTEET.....	51
	LIITTEET	54
	Liite 1. Puitekilpailutuksen ehdottomat referenssivaatimukset.....	54
	Liite 2. Sprinttien retrospektiivi –lomake.....	55
	Liite 3. Palautelomakeluonnos tiimiltä projektipäällikölle	56
	Liite 4. Palautelomakeluonnos kehitystiimiltä ohjausryhmälle	57
	Liite 5. Scrum –menetelmää varten mukautettu Trello-taulu.....	58

LYHENTEET JA TERMIT

HAMK	Hämeen ammattikorkeakoulu
IT	Information Technology
GANTT	GANTT-kaavio on tapa esittää projektin työtehtävät sekä niiden riippuvuussuhteet ja aikataulu kaavion muodossa
vaatimusmäärittely	vaatimusmäärittely on tyypillisesti dokumenttimuotoinen esitys tietojärjestelmäprojektin tavoitteista ja vaatimuksista
staging-ympäristö	tuotantokäyttöön tarkoitettua tietojärjestelmän käyttöympäristöä vastaava ympäristö, jota käytetään tuotantokäyttöön tulevien ominaisuuksien testaamiseen ja kehittämiseen
scrum	iteraatioihin perustuva ketterän projektinhallinnan ja kehittämisen menetelmä
sprint	scrum-menetelmän nimitys iteraatiolle eli tuotteen kehitysvaiheen syklille
daily scrum	sprintin jokaisena päivänä järjestettävä toteutustiimin sisäinen kokous
sprint review	sprintin tulosten esittelytilaisuus
sprint retrospective	sprintin jälkeen järjestettävä toteutustiimin sisäinen kokous, joka kehittää tiimin toimintaa
scrum master	scrum-menetelmän rooli, jonka tehtävänä on mahdollistaa scrum-tiimin toiminta niin tehokkaasti kuin mahdollista
product owner	tuoteomistaja on scrum-menetelmän rooli, joka valvoo rahoittajien etua projektissa
product backlog	tuoteomistajan priorisoima tuotteen ominaisuuslista
sprint backlog	sprintin ominaisuuslista eli lista sprintissä toteutettavista ominaisuuksista
toteutustiimi	scrum-menetelmän rooli, joka tarkoittaa niitä henkilöitä, jotka tekevät projektissa varsinaisen työn. Tästä roolista käytetään yleisesti nimeä kehitystiimi.
kompleksisuus	monimutkaisuus

Hilma

työ- ja elinkeinoministeriön ylläpitämä maksuton palvelu, jossa julkishallinnollisten hankintayksiköiden tulee ilmoittaa kansallisen ja EU-kynnysarvon ylittävät hankinnat

1 JOHDANTO

Tämä opinnäytetyö kertoo Hämeen ammattikorkeakoulun (HAMK) tietojärjestelmäprojektien hallintamenetelmien kehittämiseen liittyvistä kokemuksista. Opinnäytetyö antaa lukijalle kokonaiskuvan siitä, miten ketteriä projektihallintamenetelmiä voidaan soveltaa sellaisessa organisaatiossa, joka ei varsinaisesti ole järjestäytynyt projektien ympärille. Lukija saa myös vinkkejä siitä, miten hankintoja voidaan tehdä ketterästi, mutta kuitenkin hankintalain puitteissa.

Ketteriä projektinhallintamenetelmiä ja uudenlaista hankintatapaa pilotoitiin HAMKin verkkopalveluiden uudistamisprojektissa. Tämän projektin alkaessa Hämeen ammatillisen korkeakoulutuksen kuntayhtymä (HAKKY) käytti sekä ulkoisten että sisäisten verkkopalveluiden palvelualustana Oracle Portal -ohjelmistoa. Tämä ohjelmisto oli ollut käytössä vuodesta 2005 ja sen käyttöön sisältyi useita riskejä kuten se, ettei ympäristöä voitu enää päivittää, koska tuotteen tuki oli loppunut sekä erityisesti se, ettei ympäristöä voitu siirtää uudelle laitteistoalustalle tai edes virtualisoida.

Edellä mainituista seikoista johtuen HAMK käynnisti loppuvuonna 2012 hankkeen, jonka lopputuloksena kaikki verkkopalvelut on siirretty uudelle alustalle ja niiden sisältö ja ulkoasu on uudistettu täysin. Opinnäytetyöhön liittyvä kehittämistehtäväni oli projektin läpivienti julkisten verkkopalveluiden osalta. Esiselvitysvaiheen päätteeksi alustatuotteeksi valittiin Microsoft Sharepointin On Premise –versio eli asiakkaan omaan konesaliin asennettava versio. Yksi keskeisimmistä vaatimuksista oli se, että palvelut ovat HAMKin omassa konesalissa ja omassa ylläpidossa. Myös Sharepointin sisältö-arkkitehtuurin määrittelyn ja sisällöntuotannon tulee tapahtua omana työnä. Jo esiselvitysvaiheen aikana minulle muodostui hyvin vahva näkemys siitä, että ketterät projektinhallinta- ja kehittämismenetelmät voisivat soveltua hyvin tämän tyyppisen projektin hallintaan.

Alkuperäinen suunnitelma oli rakentaa opiskelijoiden ja henkilökunnan intrapalvelut samalle alustalle kuin julkiset verkkopalvelutkin, mutta Office 365 ja Sharepoint Online -pilvipalveluiden nopean kehityksen myötä tehtiin päätös rakentaa sisäiset verkkopalvelut edellä mainittuun pilviympäristöön ja että tämä osa eriytetään omaksi projektikseen.

Nykypäivän IT-organisaatiot kohtaavat uudenlaisia haasteita. Yhtäällä toiminnan resurssit ovat entistä niukemmat, mutta toisaalta taas organisaatioiden johto odottaa entistä parempia tuloksia ja entistä nopeammin. HAMKin tietotekniikkainvestoinnit ovat pienentyneet puolella viimeisen seitsemän vuoden aikana. Kuitenkaan palvelujen laatuvaatimukset tai odotukset esimerkiksi mobiilikäytettävyyden suhteen eivät ole laskeneet, vaan pikemminkin päinvastoin. Tämä tilanne on osaltaan ohjannut organisaatiotamme etsimään uudenlaisia, tehokkaampia menetelmiä projektien ohjaamiseen ja palveluiden kehittämiseen.

2 PROJEKTIHALLINTAMENETELMIEN LÄHTÖTILANNE HÄMEEN AMMATTIKORKEAKOULUN TIETOHALLINNOSSA

Projekti on kokonaisuus suunnitelmallista työtä, jolla on aikataulu, budjetti ja tavoitteet (Layton 2012, 9). Nykyään ”projekti”-nimi annetaan kovin herkästi lähestulkoon kaikelle työlle, vaikka kyse olisikin projektin sijaan prosessista – projektin tunnusmerkki on ainutkertaisuus ja aikarajoitteisuus, kun taas prosessi voi jatkua periaatteessa äärettömän kauan (Helsingin Yliopisto 2006). Projektille on tunnusomaista myös se, että sen parissa työskentelee sellaisia ihmisiä, jotka eivät normaalisti työskentele yhdessä vaan heidät on voitu kutsua projektiin eri organisaatioyksiköistä ja eri maantieteellisistä sijainneista (Project Management Institute 2015). Tietojärjestelmähankkeissa projektin ja prosessin raja on mielestäni vedettävissä käyttöönottohetkeen – tämän jälkeen järjestelmä siirtyy ylläpitoprosessien piiriin.

Projektinhallinnan prosessit ovat asettaminen, suunnittelu, toimeenpano, seuranta ja valvonta sekä projektin päättäminen. Projektinhallinnan osa-alueita ovat henkilöiden johtaminen, viestintä, riskien hallinta, kustannusten hallinta, laadun valvonta, laajuuden hallinta, hankinta, sidosryhmien hallinta, aikataulun hallinta ja kokonaisuuden hallinta. (Project Management Institute 2015.)

Monilla suurilla organisaatioilla on vakioitu projektinhallinnan ohjeistus tai pikemminkin menetelmä, johon kirjattuja pelisääntöjä kaikkien projektien tulee noudattaa. Entisellä työnantajallani Logica Suomi oy:llä (nykyisin CGI Suomi oy) oli hyvinkin tarkka ohjeistus, kun taas HAMKilla ei tällaista vakioitua projektinhallinnan ohjeistusta ole, vaan kukin projektipäällikkö saa toimia verrattain vapaasti, kunhan vähimmäisvaatimukset täyttyvät. Näitä vähimmäisvaatimuksia ovat projektin budjetin seuranta, riskienhallinta, aikataulun seuranta ja projektiin liittyvien töiden koordinointi.

2.1 Budjettiseuranta

Organisaatiollamme ei ole käytössä automaattista projektien kustannusten raportointia. Tämä tarkoittaa käytännössä sitä, että kunkin projektipäällikön velvollisuus on kirjata muistiin järjestelmätoimittajien laskujen summat niiden saapuessa ja asiataarkastaa laskut.

Projektipäälliköt eivät saa kirjanpitojärjestelmästä itsepalveluna raportteja projekti-kohtaisesta toteumasta. Projektipäälliköillä ei myöskään ole käytössään yhteistä mallipohjaa budjettiseurantaa varten, vaan kukin ratkaisee asian parhaaksi katsomallaan tavalla.

Tietojärjestelmäprojekteissa ei pääsääntöisesti organisaation sisäistä työtä kohdenneta projektien kustannuksiksi – ainoan poikkeuksen tähän muodostavat IT-projektipäälliköt, jotka kirjaavat työaikansa eri projekteille ja näin ollen myös heidän palkkakustannuksensa jakautuvat projekteille. Muiden kuin IT-projektipäälliköiden osalta palkkakustannusten kohdentaminen projekteille ei edes ole mahdollista, koska henkilöitä ei ole velvoitettu kohdentamaan työajan käyttöönsä millään tavalla eikä projekteissa perinteisesti ole dokumentoitu sitä, että kuka minkäkin projektin tehtävän on tehnyt.

Perinteisiä projektinhallinnan ja kehittämisen malleja käytettäessä olen kokenut budjettiseurannan kovin haastavaksi, koska toimittaja toteuttaa vaatimusmäärittelyyn kirjattuja toiminnallisuuksia omassa kehitysympäristössään, eikä valmis työ juurikaan näy asiakkaalle ennen kuin projektin loppuessa. Tällöin laskuja hyväksyttäessä on vain luotettava toimittajan kertomaan, koska laskun erittelyrivit ovat ainoa konkreettinen evidenssi tehdystä työstä.

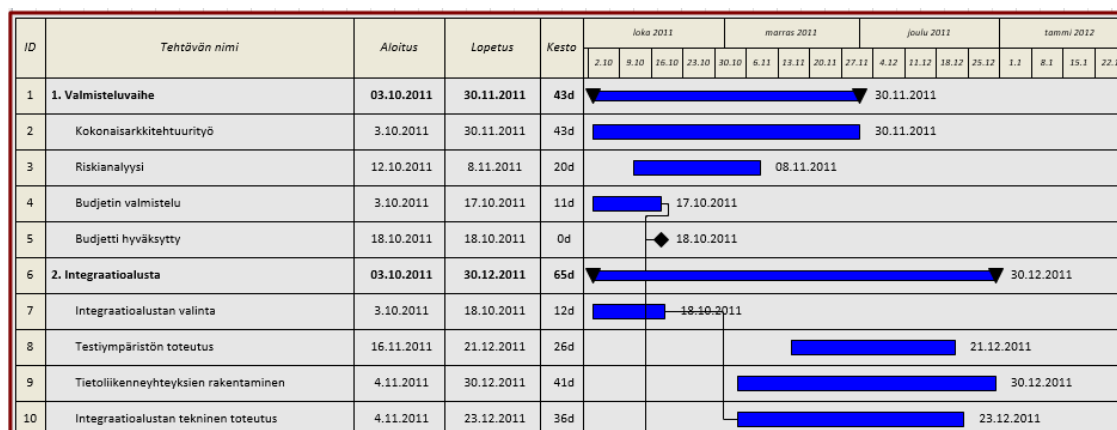
2.2 Riskienhallinta

Riskienhallinta tarkoittaa projekteissamme käytännössä sitä, että projektia käynnistettäessä tunnistetaan projektin merkittävimmät riskit ja laaditaan niistä luettelo, jossa tuodaan ilmi riski, sen toteutumisen todennäköisyys, toteutumisen vaikutukset projektiin ja mitä keinoja on riskin hallitsemiseksi. Ohjausryhmän kokouksissa projektipäällikkö esittelee projektin riskien sen hetkisen tilanteen ja ohjausryhmä päättää esityksen perusteella korjaavista toimenpiteistä.

Kokemusteni mukaan perinteisesti hallinnoituissa projekteissa riskien hallinta on ollut huomattavasti vaikeampaa. Tämä johtuu siitä, että näissä projekteissa on tyypillisesti tehty löyhempää yhteistyötä toimittajatahon kanssa, jolloin mahdollisten ongelmien esiintulossa on voinut kulua enemmän aikaa.

2.3 Aikataulun seuranta

Yleensä IT-projekteiltamme edellytetään aikataulun esittämistä jonkinlaisen aikajanan muodossa. Tätä aikataulun esitysmuotoa käytetään lähinnä ohjausryhmän suuntaan tapahtuvaan viestintään. Yleensä aikajana esitetään GANTT-kaaviona (kuva 1), koska tämä kaavio on sekä projektiryhmän että ohjausryhmien jäsenille tuttu.



KUVA 1. Erään projektin aikataulua GANTT-kaaviona

On aikajanan esitystapa mikä hyvänsä, niin sen ylläpitäminen on tyypillisesti käsityötä, koska aikajanan taustalla olevia tehtäviä ei ole kirjattu mihinkään järjestelmään. Näin ollen aikajana ei muodostu eikä päivitty automaattisesti. Lisäksi aikataulun laatiminen tapahtuu aina enemmän tai vähemmän arvaamalla, koska projektiin osallistuvien henkilöiden työkuormaa ei tiedetä, tiimin suoritustaso on arviota tehtäessä tuntematon ja projektin tavoitteetkin voivat vielä kaivata tarkentamista.

2.4 Projektin töiden koordinointi

Perinteisessä projektinhallinnassa töiden koordinoinnilla tarkoitetaan yksittäisten tehtävien ja tehtäväkokonaisuuksien tunnistamista, priorisointia, aikatauluttamista, vastuuttamista ja niiden etenemisen seuranta. IT-projekteissamme on tyypillisesti tunnistettu ja kirjattu tehtävät melko suurpiirteisesti eli kokonaisuuksien tasolla. Projektipäälliköillä itsellään on sitten saattanut olla hyvinkin seikkaperäinen lista tehtävistä. Vaatimusmäärittelyn purkaminen tehtäviksi on kuulunut projektipäällikön tehtäviin, ei niinkään työn toteuttaville asiantuntijoille tai asiantuntijatiimille. Priorisointia ei varsinaisesti ole tehty, vaan tehtävien suorittamisjärjestys on määräytynyt

lähinnä sen mukaan, mitä tehtäviä on tehtävä, jotta seuraava tehtäväkokonaisuus voidaan ottaa työn alle. Tämä on vesiputousmallin mukaan toimivissa projekteissa täysin käyttökelpoinen tapa toimia.

Yhteisten käytäntöjen puuttuessa projektien töiden kirjaamiskäytännöt vaihtelevat projekteittain. Tällöin projektipäällikön mahdollisesti vaihtuessa voi uuden projektipäällikön olla hyvin vaikeaa päästä perille projektin tehtävien tilanteesta.

3 KETTERÄT PROJEKTINHALLINTA- JA KEHITTÄMISMENETELMÄT

Ketterien kehittämis- ja projektinhallintamenetelmien lähtöpisteenä pidetään yleisesti Agile Manifestoa vuodelta 2001, mutta on syytä muistaa että Agile Manifesto ei syntynyt tyhjästä, vaan sen taustalla on akateemista tutkimusta ja käytännön kokemusta (Techrepublic 2009).

Tämän vuosituhannen alussa suoritettut tutkimukset osoittivat, että suurten tietojärjestelmäprojektien suunnitteleminen ja hallinnassa pitäminen perinteisin menetelmin on mahdotonta ja että suurin osa näin suunnitelluista ja johdetuista projekteista epäonnistuu (Highsmith 2002, 20-21). Watts Humphrey oli puolestaan jo vuonna 1995 julkaistussa kirjassaan “A Discipline for Software Engineering” todennut: ”For a new software system, the requirements will not be completely known until after the users have used it.” (Notarangelo 2010). Voidaan siis todeta, että tietojärjestelmien kehittämismenetelmät olivat tuolloin pahasti rikki ja että varsinkin vaatimusmäärittelyn lukitseminen johti (ja johtaa edelleenkin) väistämättä huonoihin lopputuloksiin. Ketterät menetelmät auttavat korjaamaan edellä mainitut ongelmakohdat kiinnittämällä erityistä huomiota käyttäjien tarpeisiin ja tarjoamalla käyttäjille mahdollisuuden tutustua kehitettävään järjestelmään mahdollisimman aikaisin.

3.1 Ketterät projektinhallintamenetelmät

Ketterä projektinhallinta on projektinhallinnan tyyli, joka tähtää nopean lisäarvon tuottamiseen, projektin tuotteen ja prosessien jatkuvaan parantamiseen, joustavuuteen, tiimin panoksen maksimaaliseen hyödyntämiseen ja hyvin testattujen, asiakkaan tarpeita vastaavien tuotteiden tuottamiseen (Layton 2012, 9). Edellä mainittujen tavoitteiden saavuttamista tukevat ketterien projektinhallintamenetelmien peruseriaatteen, joita ovat tiivis asiakasyhteistyö, päätösvaltaiset tiimit, lyhyet iteraatiot, muutosmyönteisyys, teknisen osaamisen ylläpitäminen ja keskittyminen asiakkaalle lisäarvoa tuottavien asioiden tekemiseen (Larman 2004, 29).

Merkittävin ketterien projektinhallintamenetelmien käyttöönottoa ajava voima on epävarmuustekijät. Epävarmuustekijät voidaan jakaa sisäisiin ja ulkoisiin epävarmuustekijöihin. Perinteiset projektinhallintamenetelmät ovat suurimmaksi osaksi kehitetty ja käytössä organisaatioissa, joilla on ollut mahdollisuus eliminoida liiketoimintansa ulkoiset epävarmuustekijät (käytännössä kilpailijat) sellaisena aikana, jolloin kilpailijoita on ollut vähemmän (globaalit vs. suljetut markkinat) ja näin ollen myös projektien kiireellisyydellä ei ole ollut merkitystä. Näillä organisaatioilla on ollut myös mahdollisuus rakentaa erilaisia esteitä mahdollisten uusien toimijoiden tielle. Esteet voivat olla lainsäädännöllisiä lobbauksen kautta tai vaikkapa kartellisopimuksia. Kun tällaiselle teollisuudenalalle saapuu uusi toimija, joka järkyttää vakiintunutta tilannetta, niin perinteiset projektinhallintamenetelmät osoittautuvat soveltumattomiksi (Chin 2003, 8-12).

Suomalaisessa IT-kentässä tällaisena tilanteena voitaneen pitää nykyisen CGI Suomi Oy:n ja Tiedon lähestulkoon monopoliasemaa valtion ja kuntien tietotekniikkahankinnoissa. Viime vuosituhaten lopulla Suomessa tapahtui erittäin suuria IT-alan yritysten fuusioita, joissa nykyinen CGI osti Kunnallistiedon (joka vastasi kuntien IT-kehityksestä) ja nykyinen Tieto osti Valtion tietotekniikkakeskuksen (jolla oli keskeinen asema valtionhallinnon it-palveluiden tuotannossa). Nämä fuusiot ja isoja toimijoita suosiva kilpailutuksia ohjaava lainsäädäntö loivat tilanteen, jossa näiden kahden ison toimijan ei tarvinnut kehittää liiketoimintaansa vaan projekteja voitiin viedä läpi vanhoin menetelmin, koska ulkopuolista painetta eli kilpailua ei ollut tai ainakaan kilpailijoilla ei ollut tasavertaista mahdollisuutta osallistua valtionhallinnon IT-kilpailutuksiin. (Kivekäs 2013.)

Kun julkisia hankintoja koskevaan lainsäädäntöä on muutettu ja samalla avattu markkinat myös kansainvälisille toimijoille, niin edellä mainitut isot IT-talot ovat joutuneet tarkistamaan toimintapojaan ketteryyttä tukeviksi. Hyvä esimerkki markkinoiden muuttumisesta on se, että tamperelaiset pienet ja keskisuuret ohjelmistotalot ovat tämän vuosikymmenen aikana vallanneet koko Suomen ohjelmistomarkkinat ja syrjäyttäneet perinteiset toimijat myös julkishallinnon kilpailutuksissa (Vänskä 2015).

Sellaisilla organisaatioilla, jotka operoivat hyvin voimakkaasti kilpailuilla liiketoimintalueilla on yleensä suurempi motivaatio toimia ketterästi ja nopeasti. Ketterät kehittämis-

menetelmät vaativat tuekseen sopivan projektinhallintatavan. Tällainen projektinhallintatapa tukee ketterien menetelmien perusarvoja kuten asiakaskeskeisyyttä ja kevyttä johtamistyyliä. Yksi käytetyimmistä ketteristä projektinhallintamenetelmistä on Scrum.

3.2 Scrum-menetelmän esittely

Scrum-menetelmä tarjoaa mallin ketterän projektin hallintaan. Scrum ei ota kantaa eikä ole kiinteästi yhteydessä projektissa tarvittaviin teknisiin ratkaisuihin. Näin ollen se soveltuu erityisen hyvin ketterän asiakasorganisaation käyttöön. (Poimala 2015.) Scrum voidaan myös määritellä viitekehykseksi, jossa monimutkaisia ongelmia ratkaistaan luovasti (Lekman Consulting 2014, 3).

3.2.1 Scrumin vaiheet

Projekti, jossa käytetään scrum-menetelmää, on jaettavissa neljään eri vaiheeseen (taulukko 1). Nämä vaiheet ovat suunnitteluvaihe, staging-vaihe, kehitysvaihe ja julkaisu. (Larman 2004, 113.)

Suunnitteluvaihe painottuu tuotevision laatimisen, budjetoinnin ja erilaisten demojen ja prototyyppien ympärille. Staging-vaiheessa tarkennetaan vaatimuksia ja määritellään riittävä määrä ominaisuuksia ensimmäistä kehitysiteeraatiota varten. Staging-vaihe on scrum-projektin ensimmäinen iteraatio eli sprint. Kehitysvaihe koostuu niin monesta iteraatiosta kuin tuotantokäyttöön valmiin tuotteen ensimmäisen version kehittäminen vaatii. Vaikkakin testausta tehdään kaikissa iteraatioissa, niin viimeisessä julkaisua edeltävässä iteraatiossa painotetaan erityisesti testaukseen ja laadunvarmistukseen liittyviä tehtäviä. (Larman 2004, 113.)

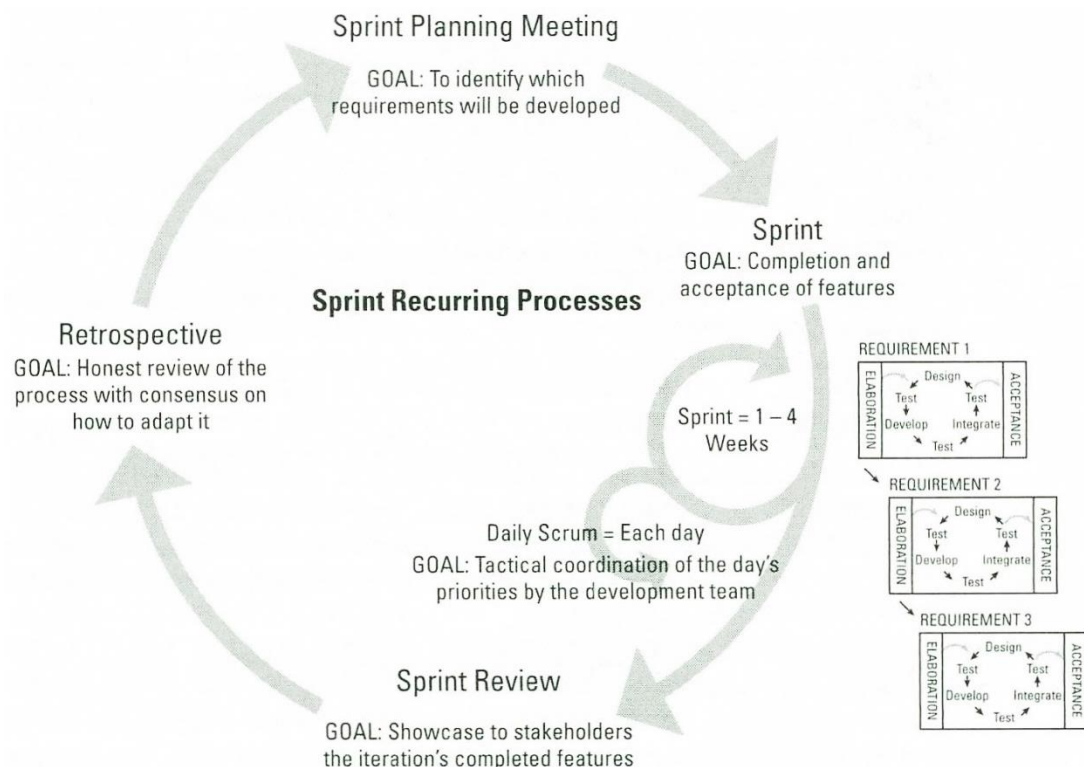
TAULUKKO 1. Scrum-projektin vaiheet (Larman 2014, 113; Layton 2012, 72)

VALMISTELUVAIHE		KEHITYSVAIHE	JULKAISU- VAIHE
Suunnitteluvaihe	Staging – vaihe		
tarkoitus: tuotevision laatiminen, rahoituksen hankkiminen	tarkoitus: vaatimusten tarkentaminen ja ensimmäisen kehitysite-raation tavoitteiden määrittely	tarkoitus: valmiin tuotteen kehittäminen useampien iteraatioiden aikana	tarkoitus: valmiin tuoteversion tuotantokäytön otto
toiminta: vision kirjoittaminen, budjetointi, product backlogin laatiminen, prototyyppien rakentaminen ja niiden testaaminen, erilaisten teknisten ratkaisujen testailu	toiminta: suunnittelutehtäviä, prototyyppien rakentaminen ja niiden testaaminen, erilaisten teknisten ratkaisujen testailu	toiminta: sprintin suunnittelu, kehitystyö ja scrum-kokoukset, Sprintin tulosten katselmointi, Sprintin retrospektiivi kehitysvaiheen toiminta eli sprint toistetaan 1-n kertaa yhden julkaisun valmistamiseksi	toiminta: dokumentointi, koulutus, myynti ja markkinointi julkaisuvaihe voi toistua useita kertoja yhden projektin aikana

Iteratiivisuus ei ole scrum-menetelmän yksinoikeus, vaan kaikki ketterät kehittämisen ja projektinhallinnan menetelmät perustuvat iteratiiviseen kehittämiseen. Scrum-menetelmässä iteratiivisuus on niin tärkeässä asemassa, että sitä jopa nimitetään scrumin luurankoksi (Schwaber 2004, 15). Luurankoa kuvaavampi suomenkielinen nimi olkoon vaikkapa kehys.

3.2.2 Sprintin suunnittelu ja työskentely sprintin aikana

Kehys (kuva 2) toimii niin, että jokaisen iteraation eli sprintin alussa kehitystiimi suunnittelee iteraatioissa toteutettavat työt sprintin suunnittelukokouksessa. Työt valitaan product backlogista, joka on tuoteomistajan ylläpitämä luettelo valmiin tuotteen ominaisuuksista. Toteutettaviksi valitaan sellaiset työt, jotka on mahdollista saada valmiiksi iteraation aikana. Sprintin työlistan englanninkielinen nimi on sprint backlog. Iteraation aikana kukin kehittäjä poimii sprint backlogilta ominaisuuksia, jotka hän toteuttaa. Iteraation aikana tiimi ohjaa työtään daily scrum-kokousten avulla. Tämä kokous järjestetään joka päivä ja sen ohjeellinen kesto on 15 minuuttia. Daily scrum -kokouksessa jokainen tiimin jäsen kertoo, mitä on tehnyt edellisen kokouksen jälkeen, mitä aikoo tehdä ennen seuraavaa kokousta ja mitä työnteon esteitä hänellä mahdollisesti on. Daily scrum -kokouksessa kukin tiimin jäsen raportoi muille tiimiläisille, ei niinkään Scrum masterille. Iteraation aikana tiimi työskentelee itsenäisesti ja esittelee iteraation lopussa sprint review -tilaisuudessa projektin omistajille aikaansaannokset. Näin projektin omistajilla on tarvittaessa mahdollisuus tehdä ajoissa tarvittavat muutokset projektiin. (Schwaber 2004, 15–18.) Sprint review-tilaisuus on ainoa scrumin tilaisuus, jossa tiimin ulkopuolisilla osallistujilla on puheoikeus – muutkin tilaisuudet ovat julkisia, mutta vain tiimin jäsenet saavat osallistua keskusteluun (Larman 2004, 119).



KUVA 2. Scrum – menetelmän kehys (Layton 2012, 72)

Iteraatio eli sprint on scrum menetelmän sydän. Iteraation aikana tiimi tutustuu vaatimukseen, perehtyy tarvittaviin teknologioihin ja arvioi omaa osaamistaan. Näiden arvioiden perusteella tiimi valitsee mielestään parhaimman toteutustavan. Tiimi uudelleenarvioi valitsemiaan toteutustapoja joka päivä ja muuttaa tarvittaessa toimintatapojaan tai ottaa käyttöön toisia teknologioita. Tämä oivaltava työskentelytapa on scrum-menetelmän sydän ja juuri se asia, joka tuottaa enemmän lisäarvoa verrattuna perinteisiin menetelmiin. (Schwaber 2004, 15, 89–90.) Sprintin aikana ei tiimi kuitenkaan saa tehdä muutoksia, jotka vaikuttaisivat sprintin tavoitteisiin tai laatutasoon heikentävästi (Lekman Consulting 2014, 7).

Sprintin maksimikestona pidetään yhtä kuukautta, koska sitä pidempi yhtäjaksoinen kehittämisjakso vaatisi ketteryuden periaatteiden vastaisesti liian tarkkaa suunnittelua ennen kehittämisen aloittamista ja lisäisi työn kompleksisuutta. Toinen mainitsemisen arvoinen syy sprintin keston rajoittamiseen on se, että projektin omistajat todennäköisesti haluavat nähdä edistymistä vähintään kuukauden välein ja mieluummin nopeamminkin. (Poimala 2015.)

3.2.3 Sprint review

Kunkin sprintin lopuksi toteutustiimillä tulee olla valmiina uutta toiminnallisuutta, joka esitellään sprintin katselmointitilaisuudessa (sprint review). Toiminnallisuus on valmis, kun se on sekä toiminnallisuudeltaan että laadultaan tuotantokäyttöön otettavissa. Keskenäistä toiminnallisuutta ei tule esitellä, koska kaikki läsnä olevat olettavat, että esiteltävänä on vain valmiita toiminnallisuuksia. Keskenäisen tuotoksen esittely olisi siis läsnäolijoiden johtamista harhaan. Artefaktien, kuten dokumenttien esittely, on sallittua vain silloin, jos ne liittyvät suoraan esiteltyyn toiminnallisuuteen. Sprinttien katselmointitilaisuudet ovat kaikille projektista kiinnostuneille avoimia tilaisuuksia. (Schwaber 2004, 116.)

3.2.4 Sprint retrospective

Heti sprintin päätyttyä järjestetään toteutustiimin sisäinen sprintin retrospektiivi, jossa tiimi analysoi omaa toimintaansa ja miettii, mitä he voisivat tehdä parantaakseen tiimensä

toimintaa (Lekman Consulting 2014, 11). Retrospektiivissä tiimi listaa sprintissä sekä hyvin että huonosti sujuneet asiat. Näiden asioiden kirjaamiseen voidaan käyttää valmiita mallipohjia, jotta kirjauksista saadaan paremmin vertailukelpoisia eri projektien välillä. Näiden asioiden pohjalta tiimi miettii, mitä tiimin toiminnassa (eli kehittämisen prosessissa) tulisi muuttaa entistäkin parempiin lopputuloksiin pääsemiseksi. Seuraavan sprintin aikana toteutustiimi sopeuttaa omaa toimintaansa asettamiensa muutostavoitteiden saavuttamiseksi.

3.2.5 Henkilöiden roolit scrum-menetelmässä

Scrum menetelmässä tunnistetaan vain kolme roolia, jotka ovat tuoteomistaja (product owner), scrum master ja toteutustiimi. Kaikki projektin vastuut jaetaan näiden kolmen roolin kesken ja projektin ulkopuolisia ovat kaikki sellaiset, joilla ei ole jotakin edellä mainituista rooleista. (Schwaber 2004, 16). Scrum-projektin ulkopuolisilla henkilöillä ei tule olla ylimääräistä vaikutusvaltaa projektiin, koska he eivät ole vastuussakaan siitä. Projektin sidosryhmien erilaisia sitoutumisasteita kuvaa hyvin vitsi sian ja kanan yhteisestä liiketoiminnasta (kuva 3).



KUVA 3. Sitoutuminen projektiin (Vizdos, 2006).

Yllä esitetty vitsi kanan ja sian sitoutumisesta liiketoimintaan on yksi yksinkertaisimmista ja mieleenpainuvimmista tavoista havainnollistaa scrum-projektin rooleja. Scrum-projektissa sikoja ovat tuoteomistaja ja scrum master, koska he vastaavat projektin lopputuloksesta. Kanoja puolestaan ovat esimerkiksi ohjausryhmän jäsenet, koska he eivät varsinaisesti vastaa projektin tuloksista, mutta voivat hyötyä niistä. (Vizdos 2006.)

3.2.6 Tuoteomistaja

Tuoteomistaja eli product owner on rooli, joka valvoo rahoittajien etua projektissa. Tuoteomistaja laatii projektin vaatimusmäärittelyn ja julkaisusuunnitelman (release plan). Scrum-menetelmässä vaatimusmäärittelyä kutsutaan nimellä product backlog eli tuotteen työlista. Tuoteomistaja vastaa siitä, että työlista on aina ajan tasalla ja että se on priorisoitu. Priorisoitu työlista takaa sen, että projektissa toteutetaan tärkein toiminnallisuus ensin. (Schwaber 2004, 16.) Julkaisusuunnitelma eli release plan kertoo toteutustiimille ja sidosryhmille sen, mitä ja milloin projektissa valmistuu. Julkaisusuunnitelma auttaa myös priorisoimaan käyttäjätarinoita ja ohjaa tiimiä käyttäjätarinoiden toteuttamistyössä. (Scrum Alliance 2012.)

Työlistan priorisointi voidaan tehdä joko asiakasvetoisesti, joka tarkoittaa sitä, että asiakkaalle eniten hyötyä tuottavat ominaisuudet toteutetaan ensin tai riskivetoisesti, joka tarkoittaa sitä, että esimerkiksi suurimmat tekniset epävarmuustekijät sisältävät ominaisuudet toteutetaan ensin (Larman 2004, 12). Parhaaseen lopputulokseen päästään käyttämällä molempia priorisointiperusteita, koska asiakas ei välttämättä pysty hahmottamaan teknisten riskien merkittävyyttä, eivätkä kehittäjät välttämättä osaa arvostaa asiakkaan tarpeita riittävästi (Larman 2004, 12).

Aiemmassa työpaikassani Logica Suomi Oy:ssä tuotekehitysprojektien tuoteomistaja oli aina tuotteen tai tuoteperheen kehittämispäällikkö, kun taas nykyisessä työssäni rooli kuuluu asiakkaan (HAMK) projektipäällikölle, joka nimetään aina lopputuloksena syntyvän järjestelmän tai palvelun omistavasta liiketoimintayksiköstä. Joissain tapauksissa tuoteomistaja voi olla myös toimittajan tekninen projektipäällikkö (Poimala 2015).

Asiakasorganisaation, kuten HAMKin, kannalta tuoteomistajan rooli on haastava, koska tuoteomistajalla on sellaisia tehtäviä, jotka ovat perinteisesti olleet ohjausryhmän tehtäviä. Tällainen aiemmin ohjausryhmälle kuulunut tehtävä on esimerkiksi toteutettavien ominaisuuksien priorisointi. Scrum-menetelmän menestyksekkäs käyttäminen kuitenkin edellyttää, että tuoteomistaja on yksi henkilö eikä ryhmä ja että koko organisaatio luottaa tuoteomistajan arviointikykyyn ja näin ollen hänen tekemiin päätöksiin (Lekman Consulting 2014, 5).

3.2.7 Scrum master

Scrum master -roolin tehtävänä on pitää huoli siitä, että projektiin osallistuvat tuntevat menetelmän riittävällä tasolla ja noudattavat scrumin periaatteita. Scrum masterin tehtävänä on myös soveltaa scrum-menetelmää juuri kyseessä olevaan projektiin ja organisaatioon sopivalla tavalla. Scrum masterin tulee osata tasapainotella sprintin tavoitteiden ja organisaation muutoksen sietokyvyn välillä. Scrum masterin auktoriteetti ei perustu organisatoriseen asemaan, vaan käytössä olevan projektihallintamenetelmän, scrumin, tuntemiseen. (Schwaber 2004, 31.)

Perinteisessä projektihallinnassa projektipäällikön tehtävät ovat usein keskittyneet tehtävien suunnittelun ja hallinnan ympärille. Scrum-menetelmässä scrum masterin ei tulisi suunnitella tehtäviä kovinkaan tarkasti etukäteen, koska tarkempi tehtävien suunnittelu kuuluu toteutustiimin vastuulle – itseohjautuva tiimi on yksi scrumin perusarvoista. Sen sijaan scrum master auttaa tuoteomistajaa valitsemaan kaikkein eniten lisäarvoa tuottavat käyttäjätarinat product backlogiin. Scrum masterin tärkein tehtävä on auttaa toteutustiimiä muuntamaan nämä tarinat käyttäjiä hyödyntäviksi toiminnallisuuksiksi. (Schwaber 2004, 40.)

Scrum masterin ja perinteisen projektipäällikön roolin suurin ero on se, että projektipäällikkö suunnittelee, organisoii ja koordinoi, kun taas scrum masterin ensisijainen tehtävä on mahdollistaa. Mahdollistamisen lisäksi scrum master johtaa inspiroimalla ja motivoimalla (The Wall Street Journal 2009). Voidaan siis todeta perinteisen projektipäällikön olevan perinteinen esimies (eng. manager) kun taas scrum master on johtaja (eng. leader) (CC Pace Systems 2003, 6).

3.2.8 Toteutustiimi

Toteutustiimi koostuu henkilöistä, jotka tekevät varsinaisen työn projektissa (Poimala 2015). Scrum-menetelmässä toteutustiimi suunnittelee itse oman työnsä sen sijaan, että projektipäällikkö organisoisi työtä. Scrum-menetelmää käyttävä tiimi työskentelee yhdessä ja yhteistyössä, sen sijaan että tiimin jäsenet työskentelisivät yksilöinä. (Schwaber 2004, 90).

Toteutustiimiä ei rakenneta perinteisen projektin tyyliin tiukasti roolittamalla eli nimeämällä kuhunkin rooliin henkilö. Rooleja ovat esimerkiksi arkkitehti, ohjelmoija ja testaaja. Scrum-menetelmässä tiimiin valitaan sopivaa osaamista omaavat henkilöt eikä heille nimetä tarkkoja rooleja. Näin edesautetaan sitä, että tiimi ottaa yhdessä kokonaisvastuun projektista eikä tehtävien pallottelua tapahdu. (Poimala 2015.) Roolien puuttuminen mahdollistaa myös tiimin jäsenten oppimisen toisiltaan, kun tiukasti rajattuja tiimin sisäisiä vastuualueita ei ole.

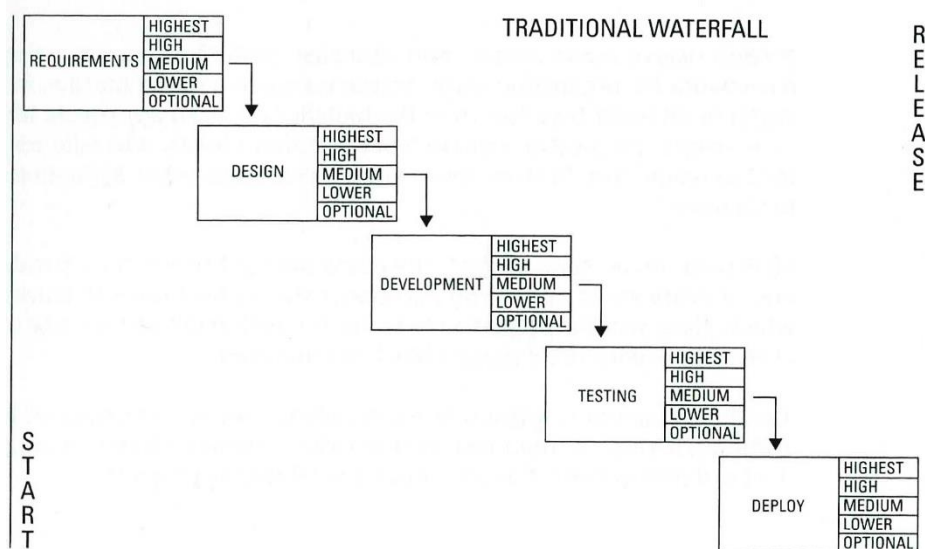
Siirtymä perinteisesti johdetuista projektiryhmän scrum-menetelmän mukaisiin toteutustiimeihin ei välttämättä ole helppo. Scrum masterin tehtävä on tukea tiimiä muutoksessa ohjaamalla heitä scrum-menetelmän käyttämisessä. Scrum-menetelmä tarjoaa tiimille useita työkaluja siirtymävaiheeseen (näitä samoja työkaluja toki käytetään jatkuvastikin). Sprintin suunnittelukokouksessa tiimi valitsee tuotteen ominaisuuslistalta tärkeimmät ominaisuudet toteutettavaksi alkavassa sprintissä. Tämä on ensimmäinen tilanne, jossa scrum -menetelmää käyttöönottava tiimi huomaa, että he ovat itseohjautuva tiimi. Toinen siirtymävaihetta merkittävästi helpottava työkalu on daily scrum -käytäntö, näissä kokouksissa tiimin jäsenet oppivat raportoimaan toisilleen – ei Scrum masterille. (Schwaber 2004, 89.) Koska kehitystiimi on itseohjautuva, on tiimin jäsenten teknisen osaamisen taso äärimmäisen tärkeää – ketterissä menetelmissä arvostetaan teknistä osaamista ja osaamisen ylläpitämiseen panostetaan (Larman 2004, 29).

Scrum-menetelmä nojaa yksilöiden väliseen, nopeaan ja epämuodolliseen viestintään. Erilaiset formaalit dokumentointi- ja kommunikointikäytännöt muodostavat esteitä toteutustiimin tehokkaalle toiminnalle (Schwaber 2004, 90). Tällaisia käytänteitä voivat olla esimerkiksi raskaat dokumentointikäytännöt tai esimerkiksi Helpdesk-tikettien tai muiden scrumin ulkopuolisten menettelyiden käyttäminen tiimiläisten tehtävien hallinnointiin. Helpdeskin kaltaisten järjestelmien käyttäminen vieraannuttaa asiantuntijat meillä olevasta projektista ja pahimmillaan aiheuttaa tehtävien pompottelua henkilöltä toiselle. Helpdesk-järjestelmien käyttö voi myös aiheuttaa sen, että tiimin itseohjautuvuuden sijaan scrum master joutuu jakamaan tehtäviä perinteiseen tyyliin kullekin tiimiläiselle. Tällöin ei käytetä scrum-menetelmää, vaan perinteisiä projektinhallinnan ja kehittämisen menetelmiä.

3.3 Ketterät menetelmät verrattuna vesiputousmalliin

Vesiputousmallia (englanniksi waterfall) käytettiin yleisesti vielä viime vuosikymmenellä. Vesiputousmalli pohjautuu toisen maailmansodan aikaisiin projektinhallintamenetelmiin, joita käytettiin Yhdysvaltain asevoimien tietokoneprojekteissa. Tuon aikakauden tietokoneprojektit keskittyivät lähinnä laitteisiin, joten vesiputousta muistuttava vaiheittainen projektimalli sai runsaasti suosiota. (Layton 2012, 10-11.)

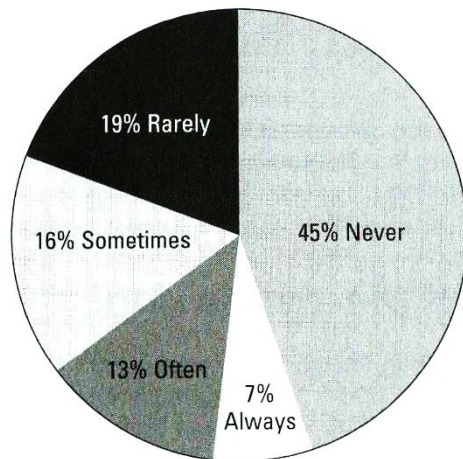
Vesiputousmallia käytettäessä jokainen projektin vaihe on saatava kokonaan valmiiksi ennen kuin voidaan siirtyä seuraavaan vaiheeseen. Kuten kuva 4 osoittaa, jokaisessa vesiputousmallin vaiheessa käydään läpi sen vaiheen työt kaikkien vaatimusten osalta, myös vähemmän tärkeiden. Ketteriä menetelmiä käytettäessä ensimmäisessä iteraatiossa toteutetaan vain kaikkein tärkeimpiä ominaisuuksia (kuva 6).



KUVA 4. Vesiputousmallin prosessi (Layton 2012).

Vesiputousmallia käytettäessä suurempi osa kustannuksista kohdistuu projektin alkupäähän kuin ketteriä menetelmiä käytettäessä. Tämä johtuu siitä, että sisällön määrittelyyn ja aikataulun suunnitteluun käytetään projektin alussa paljon aikaa, koska vesiputousmallia käytettäessä muutoksia määrittelyyn ei voida tehdä kesken projektia ilman huomattavia lisäkustannuksia. Tästä muutosten teon hankaluudesta ja kalleudesta seuraa myös se, että projektipäällikkö ja tuotteen omistaja haluavat, että toiminnallisuuksia määriteltäessä mukaan otetaan kaikki tarvittavat toiminnot, kaikki ehkä tarvittavat toiminnot, kaikki halutut toiminnot ja kaikki sellaiset toiminnot, joita saatetaan

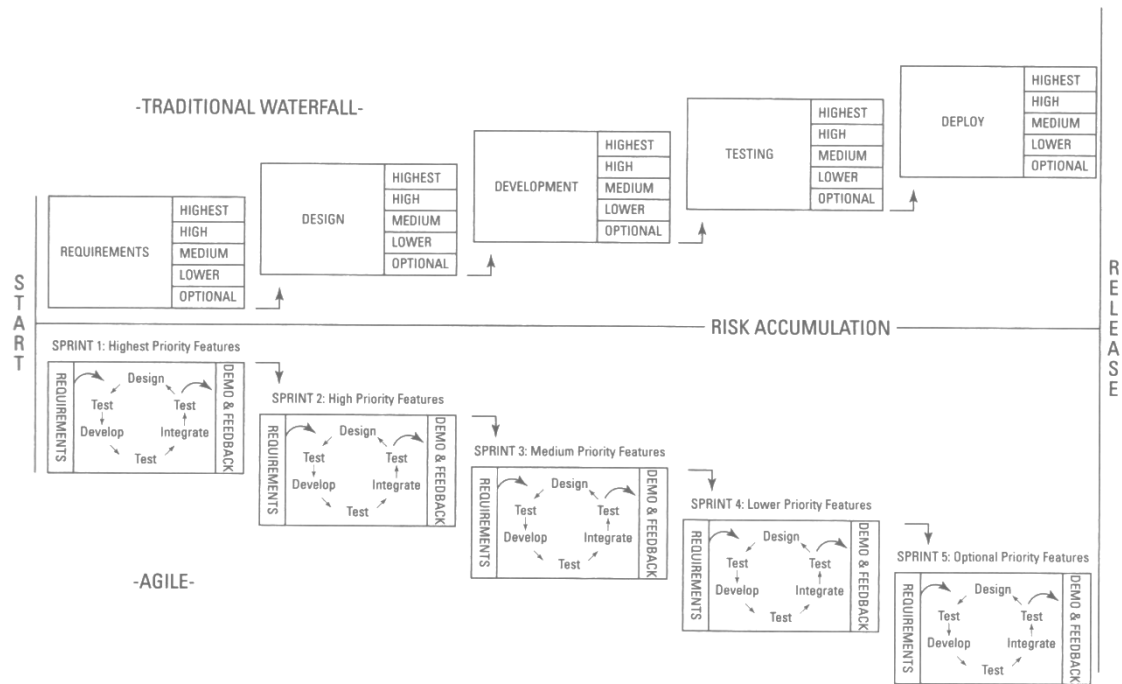
haluta. Tästä seuraa kuvassa 5 havainnollistettu projektin laajuuden hallitsematon kasvaminen täysin ja lähes turhilla ominaisuuksilla. Ketteriä menetelmiä käytettäessä kustannukset jakautuvat tasaisemmin projektin kestoajalle, koska toteutusvaiheeseen voidaan edetä paljon nopeammin ja toteutustöiden aloittaminen vaatii huomattavasti vähemmän etukäteissuunnittelua.



Copyright © 2011 by Standish Group.

KUVA 5. Turhien ominaisuuksien määrä ohjelmistoissa (Layton 2012, 12).

Kuva 6 kertoo paljon ketterien menetelmien ja vesiputousmallin välisistä eroista. Vesiputousmallia käytettäessä projektin riskit kumuloituvat projektin loppua kohti, koska projekti ei tuota käyttöönottokelpoista toiminnallisuutta ennen projektin valmistumista. Ketteriä menetelmiä käytettäessä projektin riskit pienenevät projektin edetessä, koska kukin sprintti tuottaa käyttöönotettavissa olevaa toiminnallisuutta. Jos projekti joudutaan katkaisemaan esimerkiksi ajan tai rahoituksen loppuessa, saadaan kuitenkin hyödynnettyä projektin aiempien sprinttien tuotoksia koska, kunkin sprintin lopputuloksena syntyy valmiita toiminnallisuuksia.



KUVA 6. Vesiputousmallin ja ketterien menetelmien ero riskien ja lisäarvon kannalta (Layton 2012, 16)

3.4 Ketterien menetelmien keskeisimmät hyödyt ja haitat

Eräs ketterien menetelmien haittapuoli tai pikemminkin huomioitava riski on se, että ketterät menetelmät nojaavat suoraan asiakkaan ja kehittäjän väliseen kommunikaatioon. Tästä aiheutuu se, että kehittäjän tai asiakkaan yhteyshenkilön vaihtuminen on ketterässä projektissa paljon suurempi haaste kuin vesiputousmallia käytettäessä, koska jälkimmäisessä mallissa kommunikaatio rakentuu enemmän muodollisen dokumentaation ympärille. (Udemy 2013.) Toisaalta tämä välitön ja nopea kommunikaatio myös nopeuttaa kehitystyötä.

Ketterät menetelmät tuovat selkeästi esiin sekä projektien että henkilöiden organisaatiolle tuottaman lisäarvon. Nopeasti ajateltuna tämä on ketterien menetelmien selkeä hyöty, mutta tässä piilee myös todella merkittävä riski. Perinteisessä vesiputousmallin mukaisessa, byrokraattisessa projektiorganisaatiossa on sosiaalisesti taitavien henkilöiden ollut mahdollista luovia projektista toiseen, vaikka heidän työpanoksessaan olisikin ollut toivomisen varaa. Ketterät menetelmät sen sijaan paljastavat hyvinkin nopeasti, jos jonkin tiimin jäsenen tuotos ei ole odotuksen mukainen. Ketteriä menetelmiä käyttöönnöttävän organisaation on siis syytä varautua siihen, että uusia

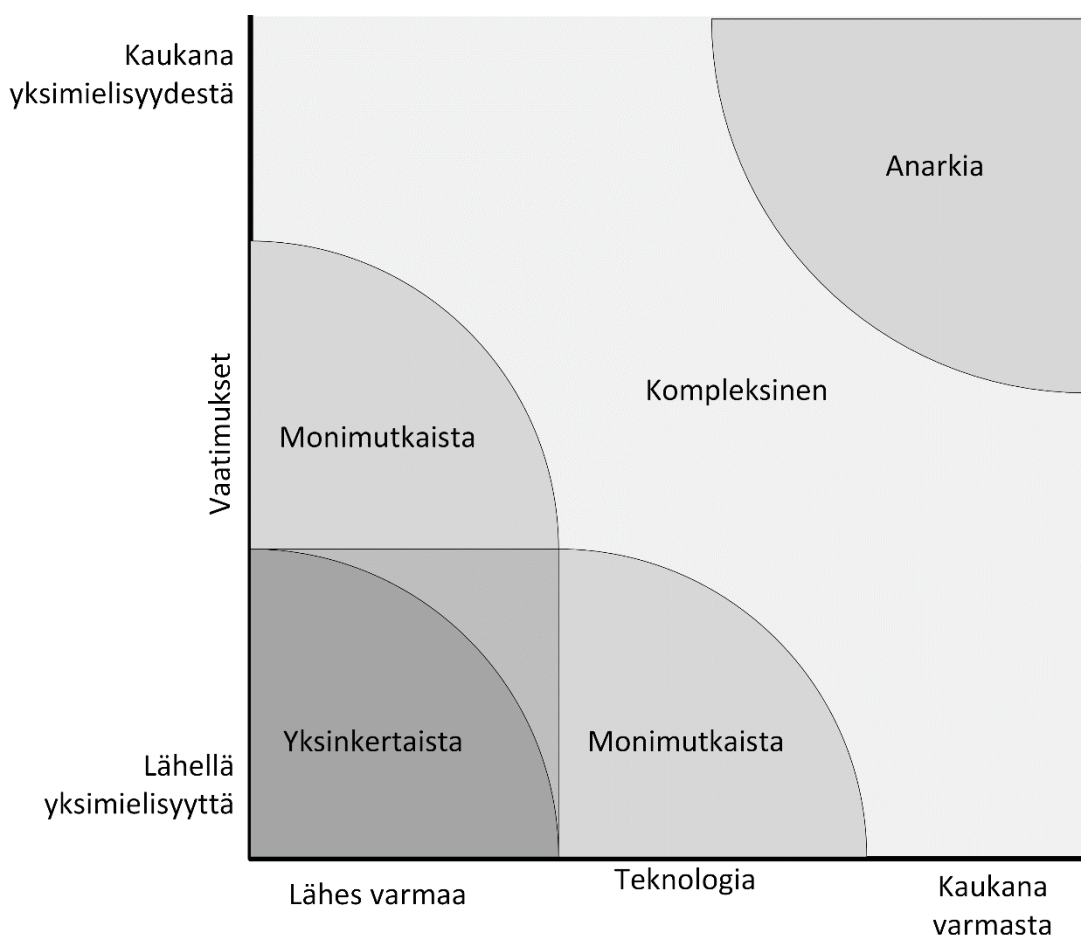
menetelmiä vastaan voidaan nousta hyvinkin rajuun vastarintaan – juuri näiden sosiaalisesti ja poliittisesti taitavien ”luovijoiden” johdolla. (InformIT 2006.)

Ketterien menetelmien hyödyt organisaatiolle ja yksilölle ovat kuitenkin haittoja suuremmat. Jo edellä esitelty tulosten ja tehdyn työn läpinäkyvyys on suuri hyöty verrattuna perinteisiin kehittämisen menetelmiin. Ketterät menetelmät mahdollistavat muutokset projektin aikana ja nämä muutokset puolestaan parantavat asiakkaan kokemaa laatua. (Scrum Alliance 2015.)

Ketterien menetelmien käyttäminen pienentää projektiin liittyviä taloudellisia riskejä, koska ketterä projekti tuottaa valmiita toiminnallisuuksia valmiiden vaiheiden sijaan (kuva 6). Ketterä projekti on myös kustannuksiltaan alhaisempi, koska siinä toteutetaan vain sellaiset ominaisuudet jotka ovat välttämättömiä tuotteen kilpailukykyisyyden kannalta (Openview Labs 2012).

4 CASE VERKKOPALVELUIDEN UUDISTAMISPROJEKTI

Olin mukana HAMKin edellisessä intra- ja internet-sivustojen uusimisprojektissa vuosina 2003-2006. Siinä projektissa noudatettiin hyvin tiukasti vesiputousmallin mukaista projektimallia. Projektissa käytettiin noin puolet ajasta vaatimusmäärittelyn laatimiseen ennen toteutustöiden aloittamista. Vaatimusmäärittely vei erityisen kauan aikaa, koska tarvittavista ominaisuuksista oli organisaatiossamme monia eri mielipiteitä. Lisäksi valittu teknologia oli erittäin keskeneräistä ja siihen liittyvää osaamista ei ollut juurikaan tarjolla. Projekti sijoittui siis kuvan 7 esittämällä kentällä ”anarkia”-lohkoon. Valtaosa käyttäjistä sai ensimmäistä kertaa tuntea uusiin palveluihin vasta tuotantokäytön alkaessa – noin kolme vuotta projektin aloittamisen jälkeen. Tästä aiheutui paljon tyytymättömyyttä ja jopa kokonaisten palveluiden hylkäämistä heti käyttöönoton jälkeen.



KUVA 7. Ohjelmistokehityksen kompleksisuus. (Schwaber 2004, 13)

HAMK aloitti julkisten ja sisäisten verkkopalveluiden uudistamisprojektin syyskuussa 2011. Minut nimettiin tähän projektiin tietohallinnon projektipäälliköksi. Projekti

aloitettiin JHS 172 -suosituksen mukaisella esiselvitystyöllä, jonka lopputuloksena Microsoft Sharepoint valittiin uusien verkkopalveluiden tekniseksi alustaksi. Esiselvitysprojekti päättyi tammikuussa 2012.

Varsinainen projekti päästiin aloittamaan syksyllä 2012 ja ensimmäinen staging-ympäristö oli verkkosivustojen kehittämisestä ja ylläpitämisestä vastaavien henkilöiden käytettävissä marraskuussa 2012. Tammikuussa 2013 minulle kävi selväksi, että emme pysty viemään projektia menestyksekkäästi läpi perinteisillä menetelmillä. Tähän vaikutti osaltaan se, että pidimme projektia kokonaisuutena erittäin monimutkaisena ja toisaalta se, että oli nähtävissä ensimmäiset merkit vaatimusmäärittelytyön venymisestä, olihan kyse sellaisesta projektista, jonka tavoitteista melkein kaikilla organisaatiossamme oli oma mielipide ja hyvin korkeat odotukset. Kuvassa 7 on havainnollistettu ohjelmistoprojektin kompleksisuutta. Tässä projektissa olimme tilanteessa, jossa olimme ottamassa käyttöön meille uutta teknologiaa liiketoimintavaatimuksiltaan hyvin vaativaan käyttötarkoitukseen. Lisäksi tunsimme aloitushetkellä vain hyvin pienen osan liiketoimintavaatimuksista. Projektin kompleksisuus oli siis korkea, eikä olisi tarvittu kovinkaan suurta muutosta vaatimuksissa tai teknologiassa, jotta projekti olisi ajautunut anarkiaan (ks. kuva 7), kuten kävi käytöstä poistuvan verkkopalvelualustan käyttöönottoprojektissa. Päätin siis yrittää vähentää projektin kompleksisuutta pilkkomalla sen pienempiin osiin käyttäen ketteriä menetelmiä. Iteratiivinen kehitys ei sinällään pienennä koko projektin kompleksisuutta, mutta se pysyy yksittäisen iteraation sisällä hallittavissa olevalla tasolla. Eräs keskeinen menestystekijä asiakasorganisaation ketterissä projekteissa on sopivan hankintamallin käyttäminen.

4.1 Ketterät menetelmät tässä projektissa

Projektissa hyödynnettiin scrum-menetelmää soveltaen. Sprinttien pituus oli projektin alkuvaiheissa kaksi viikkoa, pidensimme sprinttien keston neljään viikkoon noin puolivälissä projektia.

Projektissa ei pidetty päivittäisiä scrum-kokouksia (daily scrum), vaan nämä kokoukset järjestettiin kaksi kertaa viikossa: tiistaisin ja torstaisin. Scrum-kokousten kellonajat pysyivät koko projektin ajan vakioina.

Kunkin sprintin jälkeen järjestimme sprintin katselmoinnin (sprint review) ja retrospektiivin. Nämä tilaisuudet järjestettiin samassa yhteydessä ja niiden yhdistetty kesto oli maksimissaan neljä tuntia. Kukin sprintti suunniteltiin omassa suunnittelukokouksessaan, joiden ohjeellinen kesto oli kaksi tuntia.

Ominaisuuslista koostettiin product backlogiin, jota ylläpidettiin eri välinein projektin aikana. Sprintin ominaisuuslistaa ylläpidettiin samalla välineellä kuin product backlogiakin. Sprinttien retrospektiivien dokumentoimiseen käytettiin vakioitua dokumenttipohjaa koko projektin ajan.

4.2 Ketterää projektia tukeva hankintamalli

Sekä minä että kehitettävän verkkopalvelun omistaja olimme jo aikaisessa vaiheessa sitä mieltä, että emme voi kilpailuttaa laajoja verkkopalveluja perinteiseen tapaan yhtenä kokonaisuutena. Päädyimme tähän lopputulokseen, koska emme halunneet viedä projektia läpi perinteisen vesiputousmallin mukaisesti ja halusimme, että myös hankinnat tehdään ketteriä kehittämismenetelmiä tukevalla tavalla. Vesiputousmallin käyttäminen olisi ollut mahdotonta, koska projektia aloitettaessa meillä ei ollut selkeästi tiedossa, minkälainen lopputuloksen tulisi olla – meillä oli ainoastaan visio lopputuloksesta. Visio on yhteenveto siitä, miten ketterän projektin tuloksena syntyvä tuote tukee organisaation strategiaa (Layton 2012, 113). Hyvin laadittu ja tiivis visio voidaan avata kuulijalle hissikyödyn aikana (Moore 2006, 113), eikä visio saa olla tuotteen ominaisuuksien luettelma (Scrum Alliance 2009). Vaikka ketterät kehittämismenetelmät tukevat projektin aikana tehtäviä muutoksia, niin vision suhteen muutokset eivät ole toivottavia, vaan ne tyypillisesti aiheuttavat sekaannusta, motivaation laskua ja jopa projektin epäonnistumisen (Scrum Alliance 2009). Jos käyttöön otettava tietojärjestelmä halutaan kilpailuttaa yhtenä kokonaisuutena ja vesiputousmallia noudattavana projektina, niin silloin sen toiminnallisuuden tulee olla hyvin pitkälle määriteltä kilpailutusta tehtäessä eli pelkkä visio ei riitä. Kokemuksieni mukaan epämääräisesti toteutettu määrittelyvaihe vesiputousmallin mukaisessa projektissa johtaa väistämättä siihen, että tarjouksien summat eivät tule pitämään yhtä toteutuneiden kustannuksien kanssa. Toinen hankintatapaa ohjaava seikka oli se, että ostopalveluihin käytettävää rahaa oli melko vähän käytettävissä – toisaalta organisaation omaa työtä oli enemmän käytettävissä.

Olin jo aikaisemmin saanut tietooni, että Tieteen tietotekniikan keskus CSC oli käyttänyt tietovarastokilpailutuksessaan sellaista menettelyä, jossa he pyysivät tarjoukset asiantuntijatyön tuntihinnoista ilman kovinkaan tarkkaa määrittelyä siitä, mitä työtä tai toiminnallisuuksia aiotaan toteuttaa tai tehdä. CSC:n kokemukset olivat kuulemani mukaan olleet hyvinkin positiivisia. Päätimme hyödyntää tätä ajatusta, mutta jatkajalostimme sitä niin, että haimme asiantuntijatyön hintakilpailutuksella kolme sopimus-toimittajaa. Mielestämme tällaisen monitoimittaja-mallin avulla pystytään pienentämään vendor lock -riskiä ja lisäksi saadaan varmistettua se, että minikilpailutuksissa löydämme aina kumppanin, joka pystyy sitoutumaan kulloinkin kyseessä olevaan aikatauluun. Tämä malli myös tukee ketterää kehitysmenetelmää, koska voimme ottaa toiminnallisuuksia kehitettäväksi sitä mukaa, kun niitä saadaan määritellyksi eikä ole siis tarvetta tehdä määrittelyitä pitkälle tulevaisuuteen.

Valintaamme puoltaa myös hankinta-asiantuntijoiden näkemys siitä, että puitejärjestely sopii hyvin tämän kaltaiseen tilanteeseen, jossa hankittavien palveluiden sisältö voi olla laajakin, mutta toisaalta kokonaisuudella on selkeät rajat. Hankintojen erityisasiantuntijat ovat myös sitä mieltä, että puitejärjestely on erityisen sopiva hankittaessa asiantuntijapalveluita (Koivisto 2012).

4.2.1 Puitesopimuskilpailutus

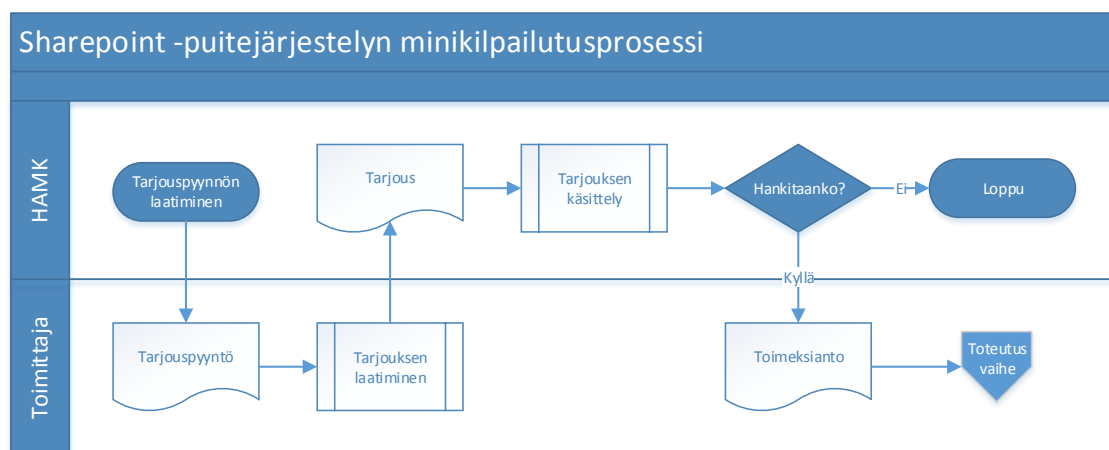
Kilpailutimme puitejärjestelyn Sharepoint-sovelluskehitystyöstä ja Sharepoint asiantuntijapalveluista. Halusimme sopimuskauden olevan kestoaltaan sellainen, että saamme sen aikana vietyä läpi sekä itse projektin että mahdollisesti tarvittavaa jatko-kehitystä. Hankintalaki (348/2007) määrää, että sopimustoimittajia tulee olla joko yksi tai kolme. Tässä oli siis selkeä riski, että emme olisi saaneet riittävästi tarjouksia kolmen sopimustoimittajan valitsemiseksi. Emme sitoutuneet hankintoihin sopimuskauteen. Tällainen kilpailutusmenettely oli myös organisaatiomme kilpailutusasiantuntijalle uutta ja tarjouspyynnön laatimiseen kului kalenteriaikaa useampi kuukausi. Erityisen työlästä oli kehittää laadullisten tekijöiden arviointia varten kriteeristö.

Tarjouspyyntö julkaistiin Hilmassa 25.7.2013 ja aikaa tarjoamiseen oli 13.8.2013 kello 16 saakka. Onneksemme riski vähistä tarjoajista ei toteutunut, vaan saimme kuusi kappaletta tarjouksia. Yksi tarjouksista toimitettiin meille eri tavalla kuin tarjouspyynnössä oli

määritetty. Tämä tarjous hylättiin, eikä sitä otettu vertailuun mukaan. Pisteitä antavia, laadullisia kriteereitä olivat referenssit määrittelemistämme osaamisalueilta. Osa referenssivaatimuksista oli ehdottomia eli tarjoajan oli täytettävä ko. ehto ollakseen kelpoinen (liite 1). Suurena yllätyksenä tuli se, että yksi tarjoajista oli kirjannut laadullisiin referensseihin sellaisen referenssin, joka ei pitänyt paikkaansa. Tästä syystä ja myös taaksemme tarjoajien tasapuolisen kohtelun, päätimme tarkistaa kaikki referenssit. Tämä tarkistustyö oli työläs vaihe, mutta ei kuitenkaan vaikuttanut tarjousten käsittelyn aika-tilaan. Puitesopimukset kolmen sopimustoimittajan kanssa astuivat voimaan 10.9.2013 ja ovat voimassa kaksi vuotta, lisäksi asiakkaalla on mahdollisuus jatkaa sopimuksia kahdella optiovuodella.

4.2.2 Minikilpailutukset

Minikilpailutuksilla tarkoitetaan valittujen puitejärjestelytoimittajien välillä suoritettua tarjouskilpailua. Minikilpailutusten tärkeimmät pelisäännöt on kirjattu puitesopimukseen. Sopimukseen kirjattiin tarjouspyyntöprosessi (kuva 8). Tärkeimmät minikilpailutuksesta sovitut asiat ovat olleet tarjouksiin vastaamisen määräajat ja se, että minikilpailutushankinnasta ei tehdä viranhaltijapäätöstä, vaan hankinnan vahvistaa asiakkaan (sopimuksessa nimetty) ostaja. Hankintojen tekeminen on onnistunut huomattavasti nopeammin, kun viranhaltijapäätöksiä ei ole tarvinnut odottaa. Halusimme näin myös varmistaa sen, että minikilpailutusten tuloksista ei tule valituksia (kun ei ole viranhaltijapäätöstä ei ole mistä valittaa). Jo puitesopimusvaiheessa minikilpailutusten pelisäännöistä sopimisen ansiota on se, että tarjoajien ei ole tarvinnut kovinkaan paljoa pyytää lisätietoja minikilpailutusten yhteydessä. Puitesopimusten voimaantulon jälkeen tehtiin jo vuonna 2013 viisi kappaletta minikilpailutuksia. Vuonna 2014 minikilpailutuksia tehtiin neljä kappaletta.



KUVA 8. Minikilpailutusprosessi

Vaikka puitejärjestelyn sisällä tehtävä minikilpailutus onkin ostajan kannalta paljon avointa tarjouskilpailua kevyempi vaihtoehto, niin siinäkin on huomioitavia asioita. Jos tarjouspyynnössä pyydetään tarjouksen kohteena olevalle työlle nimetyt tekijät, niin silloin olisi hyvä, että tarjoajilla olisi yhteinen CV-pohja käytössään. Tämä nopeuttaisi tarjousten käsittelyä. Lisäksi kun tarjoukseen on nimetty useampia tekijöitä, tulisi tarjoajaa pyytää nimeämään vastuuhenkilö – näin vältetään tilanteilta, joissa vastuut jäävät epäselviksi.

4.2.3 Kokemukset ketteristä hankinnoista

Kokemukset useamman toimittajan käyttämisestä samassa projektissa ovat olleet pääsääntöisesti hyviä. Osaltaan positiivisiin kokemuksiin on varmasti vaikuttanut se, että minikilpailukset toteutettiin kuitenkin kokonaisuuksien ympäriltä – aina kun se oli mahdollista. Esimerkiksi Sharepoint-käyttöympäristön tekninen konsultointi kilpailutettiin yhtenä kokonaisuutena, koska sen osalta tiedossa oli, minkä tyyppisiä palveluita tulemme tarvitsemaan. Näin ollen teknisiä asioita on konsultoinut vain yksi puitesopimuskumppani. Räätelöityjen sovellusten osalta taas minikilpailutuksissa on tullut valituksi aina sama puitesopimuskumppani vaikka muutkin ovat tarjouskilpailuihin osallistuneet. Puitesopimuksen voimassaolon aikana meille onkin muodostunut sellainen kuva, että puitesopimuskumppanimme ovat osaamiseltaan hyvin erilailla profiloituneita ja näin heille on löytynyt projektissamme luontevat roolit.

Mielestäni tällainen hankintatapa ei sovellu käytettäväksi, jos organisaatiolla ei ole mahdollisuutta tai halua ottaa kokonaisvastuuta hankinnan kohteesta. Tämän projektin tapauksessa kokonaisvastuu tarkoittaa sitä, että HAMK vastaa sivustojen toimivuudesta ja käytettävyydestä yksin – tosin yhteistyökumppaneiden tuella. Tämä on aiempaan verrattuna myös taloudellisempi tai ainakin kustannustehokkaampi tapa toimia, koska kiinteitä ylläpitomaksuja ei ole, vaan asiantuntijapalveluita käytetään tarvittaessa ja asiantuntijapalveluiden tuntiveloitushinta sovitaan kiinteäksi koko sopimuskaudeksi. Vanhan verkkopalvelualustan ylläpidosta maksoimme kuitenkin yli 2000 euroa kuukaudessa kiinteää ylläpitomaksua, vaikka yhteistyökumppanille kertyi tehtyjä tunteja niukasti. Minikilpailutusten käyttäminen on myös helpottanut selkeästi projektipäällikön ylläpitämää budjet-tiseurantaa, koska toimitettavat kokonaisuudet ovat pienempiä. Myös laskujen asiatar-kas-tus on mielekkäämpää, kun laskun rivejä voi peilata meille toimitettuihin, käyttövalmii-siin toiminnallisuuksiin pelkän sanahelinän sijaan. Kaiken kaikkiaan olemme kokeneet puitejärjestelyn niin hyvin toimivaksi, että jatkoimme sopimusta kahdella optiovuodella.

4.2.4 Ketterien ja perinteisten hankintojen olennaisimmat erot

Taulukkoon 2 on koottu keskeisimpiä perinteisten ja ketterien hankintojen välisiä eroja. Tässä projektissa kerätyt kokemukset ketteristä hankinnoista ja aiemmat kokemukset pe-rinteisistä hankinnoista tukevat taulukon 2 havaintoja.

TAULUKKO 2. Perinteisten ja ketterien hankintojen erot (Layton 2012, 197)

Perinteiset hankinnat	Ketterät hankinnat
Projektipäällikkö ja hankintaorganisaatio vastaavat hankin-noista	Itseohjautuva toteutustiimi tunnistaa hankintakohteet
Sopimukset sisältävät kiinteitä vaatimuksia, kattavaa doku-mentaatiota ja laajan projektisuunnitelman	Sopimukset rakentuvat rakennettavat toiminnallisuuden ympä-rille – ei dokumentaation tai projektinhallintakäytänteiden ym-pärille
Sopimusneuvottelut voivat olla hyvinkin vaikeita ja voivat va-hingoittaa asiakkaan ja toimittajan suhteita jo ennen kuin pro-jekti on päässyt alkamaan	Keskitytään positiivisen ja yhteistoimintaa tukevan asiakas-toimittaja -suhteen luomiseen ja ylläpitämiseen
Toimittajan vaihtaminen hankinnan alettua on mahdotonta (tai ainakin erittäin kallista), koska uuden toimittajan on erittäin vaikea ymmärtää valtavaa määrää kesken olevaa työtä	Toimittajan vaihtaminen kesken projektin on mahdollista, koska toimittaja työskentelee aina vain yhden sprintin tavoit-teiden toteuttamiseksi. Tämä vähentää kesken olevan työn määrää ja antaa uudelle toimittajalle mahdollisuuden aloittaa seuraavan sprintin vaatimusten työstämisen heti.

4.2.5 Ketterät hankinnat tulevaisuudessa

HAMK on perinteisesti pyrkinyt tietojärjestelmähankinnoissaan löytämään tarpeeseensa sopivia valmisohjelmistoja, joiden räätälöintitarve olisi mahdollisimman vähäinen. Useimmiten tällainen vaihtoehto onkin löydettävissä, mutta joissain tapauksissa sopivaa valmisohjelmistoa ei ole saatavissa. Tällöin on kilpailutettava asiakaskohtainen tietojärjestelmäprojekti. Olemme tällaisissa tilanteissa hakeneet kilpailutuksella aina valmista, meille kehitettyä järjestelmää räätälöintineen. Näissä projekteissa haastavaa on tarjouspyynnön liitteenä olleiden vaatimusten tulkinta ja varsinkin niiden yksityiskohdat. Tästä on syntynyt aina jossain vaiheessa projektia ristivetoa järjestelmätoimittajan kanssa. Kuten edellä on kerrottu, tässä projektissa hankinnat tehtiin puitejärjestelyn kautta. Tämä toimintatapa on osoittautunut toimivaksi, mutta muitakin tapoja ketterää kehitystä tukevien hankintojen tekemiseksi on.

Ketteriä hankintoja voidaan lähestyä myös toisenlaisella tavalla. Sen sijaan että kilpailutetaan jonkin tietynlaisen toiminnallisuuden työmäärää (hintaa), kilpailutetaan kysessä olevan toiminnallisuuden toteuttamiseen tarvittavaa osaamista eli henkilöitä. Tällainen hankintatapa soveltuisi myös HAMK:n toimintatapaan, jossa etsitään sopivia valmisohjelmistoja. Valmisohjelmiston hankinta voitaisiin kilpailuttaa omana hankintanaan ja tarvittavat räätälöinnit erikseen. Räätälöinnit voitaisiin kilpailuttaa edellä mainitulla tavalla eli esimerkiksi pyytämällä tarjous 30 henkilötyöpäivän työpanoksesta. (Toivanen 2014.) Näin toimimalla voitaisiin saada tehtyä valmisohjelmiston hankinta ilman, että hankintalain 15§ mukainen käyttöoikeussopimuksen kynnysarvo (30 000 euroa) ylittyy (TEM 2015). Jotkut ammatillisen verkostoni jäsenet tulkitsevat tämän hankintalain 20§ kielletyksi hankinnan pilkkomiseksi, mutta mielestäni tätä ei voida tulkita laissa mainituksi ”poikkeukselliseksi menetelmäksi”. Tällaisista hankinnoista voi ajatella olevan hyötyä esimerkiksi pienille yrityksille, jotka pääsevät mukaan tarjouskilpailuihin räätälöintityöstä, vaikkei heillä olisikaan tarjota varsinaista ohjelmistoa.

Osaamisen hankkiminen vaatii asiakkaalta paljon enemmän tai ainakin erilaista osaamista kuin ”avaimet käteen” – kilpailutus. Asiakkaan projektiryhmän pitää osata toimia ketterässä projektissa ja asiakkaan tuoteomistajalla tulee olla riittävä toimivalta projektissa, jotta mahdollisimman harvoin kysymyksiin tarvitaan ohjausryhmän kannanottoa. Asiakkaan projektiryhmän tulee osallistua sata prosenttisella työpanoksella projektiin, jotta ostettu kehitystiimi saa esittämiinsä kysymyksiin vastaukset säällisessä ajassa. Asiakkaan

tulee myös sisäistää se, että tällaisessa projektissa on vakioitu sekä hinta että aikataulu – laajuus sen sijaan joustaa (Toivanen 2014). Näkisin tällaisen toimintatavan hyvänä myös siksi, että sen myötä on aivan välttämätöntä pystyä asettamaan ominaisuudet tärkeysjärjestykseen. Näin toimimalla saavutetaan se, että kun projektin käytettävissä oleva aika ja raha on loppu, niin lopputulos on varmasti niin hyvä kuin mahdollista ja sisältää kaikki tärkeimmät ominaisuudet.

4.3 Kokemukset ketteristä menetelmistä

Projektiryhmälle kertyi kokemusta ketteristä menetelmistä yli vuoden ajalta, helmikuusta 2013 huhtikuuhun 2014. Projektiryhmän jäsenet haastateltiin toukokuussa 2014. Haastattelu toteutettiin teemahaastatteluna, jonka keskeiset teemat olivat: alkutunnelmat, kokemukset scrum-kokouksista, kokemukset tehtävien hallinnoinnista ja valvonnasta, minkälainen kokonaisvaikutelma ketteristä menetelmistä syntyi ja mitä tekisit nyt toisin. Teemahaastattelu ei ole yhtä muodollinen kuin lomakehaastattelu, mutta kuitenkin muodollisempi kuin avoin haastattelu. Teemahaastattelu ei etene yksityiskohtaisten, valmiiksi pureskeltujen kysymysten kautta, vaan vapaammin kohdentuen ennalta valittuihin teemoihin. Teemahaastattelu on määrämuotoisempi kuin avoin haastattelu, sillä siinä valmistellut aihepiirit (teemat) ovat kaikille haastateltaville samoja, vaikkei ennalta määritettyä etenemisreittiä olekaan. (Saaranen-Kauppinen & Puusniekka 2006.)

Haastatteluita tehdessäni huomasin, että teemahaastattelu antaa haastattelijalle hyvät mahdollisuudet reagoida - jotkut haastateltavat saattavat kaivata vaikkapa enemmän ”lämmittelyaiheita”, jotta he rentoutuvat ja intoutuvat vastaamaan avoimesti. Haastattelin neljä henkilöä eli koko projektiryhmän (eli tämän projektin scrum-tiimin) ja kunkin henkilön haastattelu kesti 15–25 minuuttia. Kaikki haastateltavat ovat työskennelleet yli kymmenen vuoden ajan erilaisten IT-projektien parissa ja kaikki ovat osallistuneet useisiin vesiputousmallin mukaisesti toteutettuihin tietojärjestelmäprojekteihin.

Haastatteluiden ääniraita tallennettiin digitaaliseen muotoon litterointia, analysointia ja arkistointia varten. Haastatteluiden litterointi on tehty puhekielellä ja analysointimenetelmänä käytettiin teemoittelua. Analysoinnin yhteydessä ilmeni, että aineisto on melko kyläntynnyttä eli viimeinen haastateltava ei enää juurikaan tuonut ilmi uusia asioita.

4.3.1 Toteutustiimin kokemukset

Projektiryhmän alkutunnelmat ja ennakko-odotukset ketterien menetelmien suhteen olivat varovaisen positiiviset. Yksi projektiryhmäläinen ilmaisi epäilleensä alussa itse menetelmien toimivuutta, kun taas toinen epäili sitä, että miten menetelmät sopivat HAMK:n muuhun toimintaan. Yksi projektiryhmäläisistä otti uudet toimintatavat kiitollisena vastaan, koska kokemukset perinteisistä menetelmistä olivat olleet negatiivisia:

Koska mulla oli kokemusta siitä aiemmasta portaaliprojektista niin otin erittäin iloisena vastaan toisenlaisen toimintatavan. Koska mä olin nähnyt että mitä sillä vanhalla vesiputousmallilla saadaan aikaiseksi taikka että minkälainen se on menetelmänä tässä organisaatiossa, näillä toimijoilla, tollasella kohteella. Ja sehän oli valtavan pitkä, raskas - siinä mielessä niinkun vaikee projekti että kun siinä ruvettiin tekeen niin valtavan kokoista asiaa ja sitten tavoitteet muuttu matkan varrella, ihmiset vaihtu matkan varrella, maailma muuttui, tekniikka kehittyi ja niin edelleen. Niin mulla oli semmonen mielikuva, että se ei välttämättä ole se kaikista parhaan lopputuloksen aikaansaamiseksi se kaikista paras malli.

Eräs keskeisistä oivalluksistamme oli käyttäjätarinoiden, teknisten tehtävien ja virheiden hallinnointi tehtävälistojen avulla. Koin tämän tärkeäksi, jotta projektin edistymisestä on jatkuvasti oikea kuva ja jotta projektin toiminta olisi mahdollisimman läpinäkyvää. Käytimme tehtävien hallintaan alkuvaiheessa Microsoft Excel -työkirjaa, mutta siirryimme melko nopeasti käyttämään Asana-nimistä projektinhallinnan pilvipalvelua. Asana on helppokäyttöinen ja nopeasti käyttöönotettava palvelu, joka sitoo yhteen tehtävät ja niihin liittyvät keskustelut ja liitetiedostot. Tehtävälistan ylläpidosta vastasivat allekirjoittanut ja tuoteomistaja yhdessä, mutta toteutustiimin jäsenet tarkensivat ja purkivat tarinatason kuvaukset tarkemmiksi tehtäviksi ja alitehtäviksi. Projektiryhmä piti aktiivista tehtävien hallinnointia ja valvontaa hyvänä asiana. Projektiryhmä koki, että tehtävien yhteisöllinen suunnittelu ja niiden näkyminen toisille tuo lisäarvoa myös heille. Yksi haastatelluista toi esiin sen, että tehtävät kirjaamalla saadaan näkyville tehty työmäärä, vaikkei jostain syystä näkyviä tuloksia vielä olisikaan. Omalta osaltani voin todeta, että projektipäällikkönä minun velvollisuuteni on pitää huolta siitä, että projektin tehtävät ovat dokumentoitu niin, että projektipäällikön vaihtuessa uusi projektipäällikkö pääsee mahdollisimman nopeasti liikkeelle.

Koska organisaatiossamme kaikilla henkilöillä on työn alla enemmän kuin yksi projekti, päätimme alusta lähtien, että emme pidä päivittäisiä scrum-kokouksia vaan että kokoukset pidetään kaksi kertaa viikossa. Scrum-kokouksen ohjeellinen kesto oli puoli tuntia. Tämäkin kokousrytmi oli selkeä muutos aikaisempaan, koska yleensä HAMKin projekteissa on ollut tapana järjestää projektiryhmän kokous enintään kerran kuukaudessa. Projektiryhmäläiset kokivat tiheän kokousrytmin monella tapaa positiivisena asiana. Usein toistuvat kokoukset irrottavat henkilöt muista töistä ja tuovat heidät taas tähän projektiin mukaan. Kokouksien koettiin lähentäneen projektiryhmää ja terävöittäneen töiden tekemistä:

Tohon mä lisäisin vielä sen, että mun mielestä scrum -kokoukset ovat saaneet projektiryhmää läheisemmäksi. Kun usein nähtiin, käsiteltiin asiat ja ilmapiirikin muuttui rennommaksi niin mun oli jatkossa helpompi sanoa – –, tällainen tehtävä - voitko tehdä?". Tavallaan se vaikuttaa myös siihen henkin, sen lisäksi että työn hallinta helpottaa se vaikuttaa myös siihen että projektiryhmäläisillä on kivempi ilmapiiri ja rennompi ja tavallaan saa oppia tuntemaan paremmin toista kun olet niin usein sen kanssa tekemisissä. Lyhyt aikahan se on mutta kuitenkin. Tavallaan se kuitenkin pistää sut tsemppaamaan että " hei, huomenna on se scrummi, koitetaan tehdä roikkuvat tehtävät pois".

Kokoukset antoivat projektiryhmäläisille kokonaiskuvan projektin etenemisestä ja auttoivat näin kunkin oman ajankäytön suunnittelemisessa antamalla selkeän kuvan siitä, mitä muut tekevät. Näin ollen ryhmäläisillä oli parempi tieto siitä, milloin kunkin työpanosta tarvitaan. Tämä on erityisen tärkeää, kun projektiryhmän jäsenillä on muitakin töitä hoidettavanaan.

Eniten parannettavaa projektiryhmä löysi ohjausryhmätyöskentelyn tehostamisesta. Projektiryhmä koki, ettei ohjausryhmä ollut riittävän syvällä projektin työssä. Parhaana vaihtoehtona pidettiin kuitenkin sitä, että perinteisiä ohjausryhmän tehtäviä siirrettäisiin mahdollisimman paljon projektiryhmän ja tuoteomistajan vastuulle. Ohjausryhmän tehtäviksi jäisivät näin ollen vain aidosti suuret linjaukset kuten riskienhallintaan, budjetointiin ja resursointiin liittyvät tehtävät. Yksi projektiryhmän jäsen esitti tässä yhteydessä erin-

omaisen idean siitä, että projektin kriittiset viikot pyhitettäisiin 100 prosenttisesti projektin tekemiselle. Tällaisia kriittisiä viikkoja voisivat olla vaikkapa julkaisujen tuotantoon viennit.

Kaiken kaikkiaan projektiryhmälle jäi ketteristä menetelmistä positiivinen kokonaisvaikutelma ja heidän mielestään projekti onnistui näin toteutettuna vähintään yhtä hyvin kuin aikaisemmatkin. Erityisen positiivisina asioina esiin nostettiin julkaisujen käyttäminen ja läpinäkyvyys.

4.3.2 Toteutustiimin kehittyminen ketterien menetelmien hyödyntäjänä

Projekti jakaantui 16 sprinttiin, joista jokaisen jälkeen järjestettiin sprintin retrospektiivi. Retrospektiivit järjestettiin mahdollisimman pian kunkin sprintin päättymisen jälkeen. Retrospektiiveissä käytiin läpi sprintin onnistumiset ja epäonnistumiset vakioimuotoisen dokumentin pohjalta (liite 2).

Ensimmäisten neljän sprintin retrospektiiveissä tiimi vastasi lomakkeen kysymyksiin hyvinkin niukkasanaisesti ja suurin osa kommenteista liittyi ketteriin menetelmiin – tämä on juuri retrospektiivien tarkoitus. Projektin edetessä kirjaukset alkoivat muuttua yksityiskohtaisimmiksi ja kommentit liittyvät enemmän tehtäviin ja niiden suorittamisessa ilmenneisiin ongelmiin kuin ketteriin menetelmiin. Tosin aivan viimeisissäkin retrospektiiveissä löydettiin kehitettävää ketterien menetelmien soveltamisessa. Painopiste kuitenkin selkeästi siirtyi hyvin detaljitaseeseen ongelmien ja onnistumisten kirjaamiseen.

Retrospektiividokumenteista on selkeästi tunnistettavissa projektin hankalimmat vaiheet – niiden sprinttien retrospektiiveissä on kirjattu vähän onnistumisia ja paljon parannettavaa. Vastaavasti näitä ”kriisisprinttejä” seuraavien sprinttien retrospektiiveissä tilanne oli pikemminkin päinvastainen.

4.3.3 Omat kokemukseni ja mitä tekisin toisin

Projektin alkuvaiheessa minulla oli melko pintapuolinen käsitys scrum-menetelmästä ja ketteristä menetelmistä ylipäätään. Olin edellisessä työssäni saanut perustason kurssituksen scrum-menetelmän käyttämisestä ja olin toiminut scrum-menetelmää soveltavassa toteutustiimissä jäsenenä. Kynnys lähteä pilotoimaan uudenlaisia projektinhallinnan menetelmiä oli mielestäni kuitenkin matala, koska organisaatiossamme suhtaudutaan positiivisesti uusien menetelmien kokeiluihin ja toiminnan kehittämiseksi on johdon tuki olemassa.

Erityisen tyytyväinen olin scrum-kokousten vaikutukseen projektiryhmän sitoutuneisuuteen. Aiemmissa projekteissa projektiryhmän rooli on ollut kovin etäinen, eikä sitoutumisaste ole aina välttämättä ollut paras mahdollinen. Tämä on täysin normaalia ja odotettavaa, kun projektiryhmä kokoontuu vain kerran pari kuukaudessa ja projektipäällikkö käskyttää työtehtävät valmiiksi pureskeltuina. Kaksi kertaa viikossa järjestetyt scrum-kokoukset pitivät toteutustiimin jäsenet hyvin mukana projektissa eikä edellä mainitun kaltaista vieraantumista päässyt tapahtumaan. Lisäksi kokouksissa tuli viipymättä ilmi mahdolliset työnteon esteet, kuten muut projektit tai yllättävän paljon aikaa vievät ylläpityt. Scrum-kokousten keston ja sisällön suhteen olisi ollut huomattavastikin parannettavaa: kokoukset venyivät pahimmillaan jopa parin tunnin mittaisiksi puolen tunnin sijaan ja niissä käsiteltiin asioita liian yksityiskohtaisesti.

Käyttäjätarinoiden, teknisten tehtävien ja virheiden kirjaamiskäytännöt toimivat pääsääntöisesti hyvin. Joissakin tapauksissa toteutustiimin jäsenet purkivat täysin omatoimisesti monimutkaisetkin käyttäjätarinat tehtäviksi. Näin olisi varmasti tapahtunut aina, jos olisin osannut sitä heiltä vaatia ja toisaalta antanut heidän tehdä sen. Sorruin siis liian usein perinteiseen projektipäällikön rooliin ja syötin toteutustiimille valmiiksi pureskeltuja, kovin yksityiskohtaisesti kuvattuja tehtäviä – tämä on perustavaa laatua oleva virhe scrum-menetelmän implementoinnissa (Larman 2004, 127). Yksi scrumin perusarvoista on tiimin itseohjautuvuus ja sitä kautta tapahtuva oppiminen ja kehittyminen (Schwaber 2004, 90). Jotta tätä kehitystä voi tapahtua, on projektipäälliköstä tultava scrum master, pomosta johtaja. On myös muistettava, ettei yksittäinen projektipäällikkö tai scrum master voi mitenkään tietää parasta tapaa kunkin käyttäjätarinan muuttamiseksi toiminnallisuuksiksi, vaan paras ratkaisu syntyy, kun asiantuntijoista koostuva toteutustiimi etsii itse ratkaisutavan. Uskon syyn siihen, että määrittelin toteutustiimin tehtäviä liian tarkkaan olleen se,

että minulla oli projektissa myös teknisen asiantuntijan rooli scrum master -roolin lisäksi. Scrum master voi toki toimia teknisenäkin asiantuntijana ja auttaa toteutustiimiä omalla asiantuntemuksellaan, mutta vain silloin kun tiimi pyytää häneltä apua (The Agile School, 2012).

Tällä hetkellä meneillään olevissa projekteissa minun on paljon helpompi antaa toteutustiimin etsiä parhaat ratkaisut, koska kyseessä olevat toteutusteknologiat eivät kuulu omaan osaamiseeni ja voin siis keskittyä tärkeimpiin tehtäviini eli toteutustiimin auttamiseen projektin tavoitteiden saavuttamisessa. Toki tähän vaikuttaa sekin, että nyt ymmärrän teknisiin yksityiskohtiin sekaantumisen olevan scrumin periaatteiden vastaista. Näiden kokemusten perusteella pidänkin selvänä riskinä sitä, että scrum masterilla on kyseessä olevaan toteutukseen liittyvää syvällistä teknistä tuntemusta tai että hänellä on rooli myös toteuttajana. Tosin tämän riskin merkittävyys on suuresti riippuvainen siitä, kuinka hyvin scrum masterina toimiva henkilö osaa olla tuputtamatta omaa osaamistaan tiimille. Joka tapauksessa tässä projektissa onnistuimme aiempia projekteja paremmin projektin tehtävien kirjaamisessa. Tämän ansiosta projektipäällikön tai muun avainhenkilön vaihtuminen ei olisi ollut vaikutuksiltaan niin negatiivista kuin aiemmissa projekteissa.

Ohjausryhmän toiminta oli samankaltaista kuin aiemmissakin, perinteisin projektinhallintamenetelmin ohjatuissa projekteissa. Tästä seurasi se, että sprinttien katselmointikouksissa ja ohjausryhmän kokouksissa käsiteltiin osin päällekkäisiä asioita – molemmissa esiteltiin valmistuneita ominaisuuksia. Mielestäni ohjausryhmä voitaisiin kutsua sprinttien katselmointitilaisuuksiin paikalle, siellä he saisivat aina oikean kuvan töiden edistymisestä ja voisivat kommentoida uusia ominaisuuksia heti tuoreeltaan. Tämä tietenkin vaikuttaisi ohjausryhmien kokoon, kovin isoa ryhmää ei saada neljän viikon välein (eli kunkin sprintin päätteeksi) koolle. Vaihtoehto koko ohjausryhmän läsnäololle olisi vain osittainen edustus – esimerkiksi niin, että paikalla olisi ohjausryhmän puheenjohtaja ja kehittämisen kohteen omistajan edustaja.

Projektin toteutusvaiheen alkaessa esittelin tuoteomistajalle ja toteutustiimille vaiheittaisen käyttöönoton mahdollistavan release- eli julkaisumenettelyn. Tuotteen julkaisumenettely tarkoittaa sitä, että tuoteomistaja määrittelee milloin ja missä laajuudessa projektin tuloksia otetaan käyttöön. Aiemmin projekteissamme tulokset on

otettu käyttöön vasta, kun kaikki projektin työt ovat olleet valmiina. Tällä kertaa tuoteomistaja päätti, että verkkopalveluiden uudistaminen jaettiin viiteen eri julkaisuun. Ensimmäinen julkaisu oli Hämeen ammatti-instituutin julkiset verkkosivut siinä laajuudessa kuin oli käyttöönoton kannalta välttämätöntä. Toiminnallisuutta ja sisältöä laajennettiin seuraavassa julkaisussa. HAMKin julkiset verkkopalvelut julkaistiin kolmena eri julkaisuna, joista ensimmäisen ja viimeisen aikaväli oli yli vuosi. Julkaisumenettely toimi sekä minun, että tuoteomistajan mielestä hyvin. Se auttoi priorisoimaan vaatimuksia ja auttoi myös karsimaan turhia vaatimuksia pois. Julkaisujen suunnittelu olisi kannattanut aloittaa heti projektin alussa, näin meillä olisi ollut heti selkeät aikataululliset tavoitteet ja aikataulu olisi ollut helposti viestittävissä sidosryhmille.

Projektin alkuvaiheessa pyrimme arvioimaan käyttäjätarinoiden vaikeutta ja sitä kautta työmääriä. Käytimme vaikeusasteen arvioinnissa Fibonaccin lukuja (8, 16, 32, 64, 128), joita annettiin käyttäjätarinoille ja käyttäjätarinoista johdetuille tehtäville. Lukuja käytettiin niin, että mitä suurempi luku, sitä vaikeampi tehtävä. Hyvin nopeasti kävi selväksi se, että emme juurikaan osaa arvioida käyttäjätarinoiden vaikeusasteita, koska lähes kaikki käyttäjätarinoista johdetut tehtävät sisälsivät uuden oppimista. Täysin uusien tehtävien vaikeusasteen arviointi on lähes täysin arvailua ja siksi lopetimme vaikeusasteiden ja työmäärien arviointiyritykset muutaman sprintin jälkeen. Projektin etenemisen seurannan ja riskienhallinnan kannalta olisi ollut hyvä, että tehtävien vaikeusasteiden (ja samalla työmäärien) arviointi olisi ollut mahdollista – tämän avulla meille olisi verrattain nopeasti muodostunut käsitys siitä, kuinka monta vaikeuspistettä tiimi pystyy sprintissä suorittamaan. Tästä olisi suurta apua julkaisujen laajuutta määriteltäessä.

5 TOIMIVIA KÄYTÄNTEITÄ HÄMEEN AMMATTIKORKEAKOULULLE

Tässä projektissa kertyneiden tietojen ja kokemusten perusteella ketterät projektinhallinnan ja kehittämisen menetelmät voivat tulevaisuudessa olla HAMK:n pääasiassa käytettävät menetelmät. HAMK:n ei kuitenkaan kannata hylätä täysin perinteisiä projektinhallintamenetelmiä, koska niille edelleenkin löytyy erinomaisen sopivia käyttökohteita sellaisista projekteista, joiden vaatimukset ovat hyvin tiukasti rajatut ja selvillä jo ennen projektin alkua. Tällaisina projekteina voidaan pitää myös räätälöimättömien valmisohjelmistojen käyttöönottoprojekteja. Seuraavassa esitetään sellaiset peruseriaatteen, joita HAMK:n ketterissä projekteissa tulisi noudattaa.

5.1 Ketterä kilpailuttaminen

Kilpailutuksissa tulee suosia ketteriä menetelmiä hyödyntäviä järjestelmätoimittajia ja vaatia, että toimittaja hyödyntää kehitystyössään ja projektinhallinnassa ketteriä menetelmiä. Tarjouspyynnöissä tulee huomioida tämä kirjaamalla edellä mainitut vaatimukset tarjouskelpoisuuden vaatimuksiksi.

Olisi hyvä myös arvioida sitä, voidaanko kilpailutusvaiheessa kilpailuttaa valmiin ratkaisun sijaan pelkkää toteutustyötä. Tämä asettaa asiakasorganisaatiolle enemmän vaatimuksia esimerkiksi projektinhallinnan ja oman asiantuntemuksen suhteen, mutta on aidoisti ketterä tapa toteuttaa tietojärjestelmiä.

Valmisohjelmistojen hankittaessa HAMK voisi kilpailuttaa erikseen käyttöönottoprojektin ja räätälöinnit. Räätälöintien eriyttäminen muusta hankinnasta mahdollistaa nopeamman etenemisen käyttöönoton kilpailutuksessa, koska käyttöönoton tarjouspyyntövaiheessa ei tarvitse määritellä räätälöintejä.

5.2 Projektien priorisointi

Eräs ketterien menetelmien suurimmista hyödyistä verrattuna perinteisiin menetelmiin on se, että projektien kesto kalenteriajassa mitattuna on tyypillisesti lyhempi kuin perinteisissä projekteissa. Tämän hyödyn konkretisoituminen edellyttää sitä, että tiimi voi keskittyä yhteen projektiin kerrallaan. Koska HAMKissa projektien parissa työskentelee melko rajoitettu määrä henkilöitä, tällä hetkellä monet heistä työskentelevät monien projektien parissa yhtä aikaa. Tällainen työn organisointi on ketterien menetelmien vastaista ja estää projektien nopeaa edistymistä. Yksi ratkaisukeino olisi priorisoida projektit niin, että niitä tehdään yksi kerrallaan. Tällöin vain yksi projekti kerrallaan edistyisi, mutta projektien läpimenoaikaa saataisiin lyhennettyä merkittävästi.

Toinen merkittävä yksittäisen kehittäjän projektiin keskittymistä haittaava seikka on se, että projektien lisäksi kehittäjän toimivat jo olemassa olevien järjestelmien ylläpitäjinä ja heillä on myös jo käytössä olevien järjestelmien ylläpitoon ja tukipalveluihin liittyviä velvoitteita. Tätä tilannetta voitaisiin korjata esimerkiksi niin, että edellä mainitut ylläpitotehtävät olisivat kiertäviä eli kehittäjät olisivat vuoroin projektien ja vuoroin ylläpito-työiden käytettävissä – mutta eivät molempia yhtä aikaa.

5.3 Tuoteomistajan ja tiimin voimaannuttaminen

Ketteriä menetelmiä käytettäessä kutsumme projektiryhmää kehitystiimiksi ja jokaista ryhmän jäsentä kehittäjäksi. Kehitystiimillä tulee olla täydellinen määräysvalta toteutuksen yksityiskohtien suhteen. Kehitystiimin ulkopuolelta ei saa tulla minkäänlaisia vaatimuksia toteutuksen teknisten yksityiskohtien suhteen. Meidän tulee luottaa kehitystiimin asiantuntemukseen, koska vain he voivat tietää miten ongelmat tulee ratkaista. Lisäksi kehitystiimin sisällä tapahtuva luova ongelmanratkaisu kehittää tiimin jäseniä aivan eri lailla kuin valmiiksi pureskeltujen tehtävien rutiininomainen suorittaminen. Tällainen aito tiimityöskentely myös edesauttaa hiljaisen tiedon siirtämistä ja helpottaa siten organisaation henkilöiden sijaisjärjestelyitä.

Kullakin projektilla tulee olla vain yksi tuoteomistaja. Tuoteomistajan tulee olla henkilö, ei esimerkiksi henkilöryhmä kuten perinteinen projektin ohjausryhmä. Tuoteomistaja ottaa huomioon projektin ohjausryhmän toiveet esimerkiksi projektin kehitysjonon suhteen,

mutta kukaan muu kuin tuoteomistaja ei saa vaikuttaa siihen, mitä ominaisuuksia toteutustiimi kulloinkin kehittää.

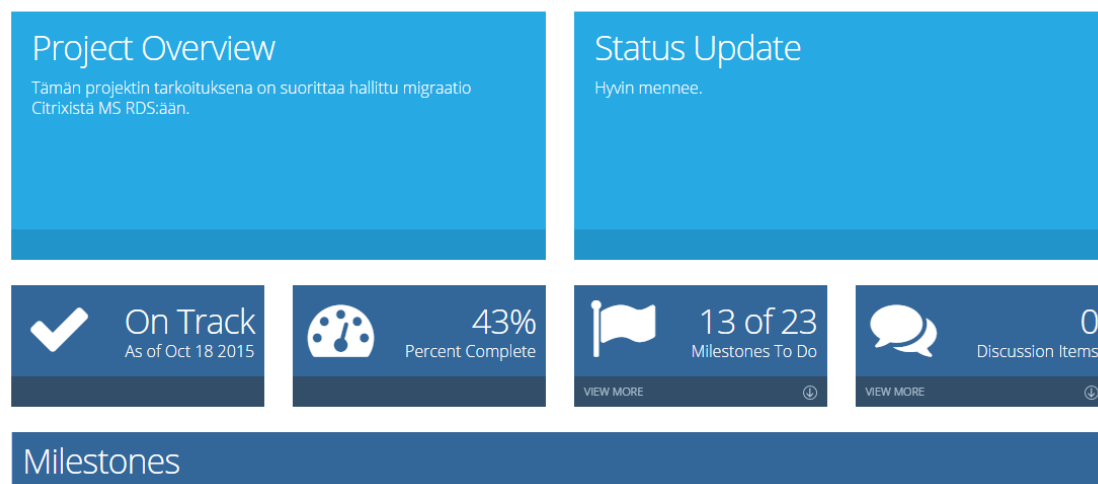
5.4 Kehitystiimin työn organisoiminen

Kehitystiimin työn organisoimiseen käytetään sovellettuna scrum-menetelmää. Puhtaasta scrum-menetelmästä HAMKin soveltama scrum eroaa siinä, että daily scrum -kokouksien sijaan kehitystiimi ja scrum master kokoustavat kaksi kertaa viikossa. Tämä myönnytys on tehtävä siksi, että scrum master -roolissa toimivilla henkilöillä on useita projekteja ja muitakin, projektien ulkopuolisia, työtehtäviä.

Iteraatioiden eli sprinttien kesto HAMKin projekteissa on neljä viikkoa. Sprintin päättymisen jälkeen sprintin tulokset esitellään kaikille avoimissa katselmoinneissa, joihin osallistuu minimissään kehitystiimi (mukaan lukien scrum master) ja tuoteomistaja. Katselmointitilaisuuden ohjeellinen kesto on neljä tuntia.

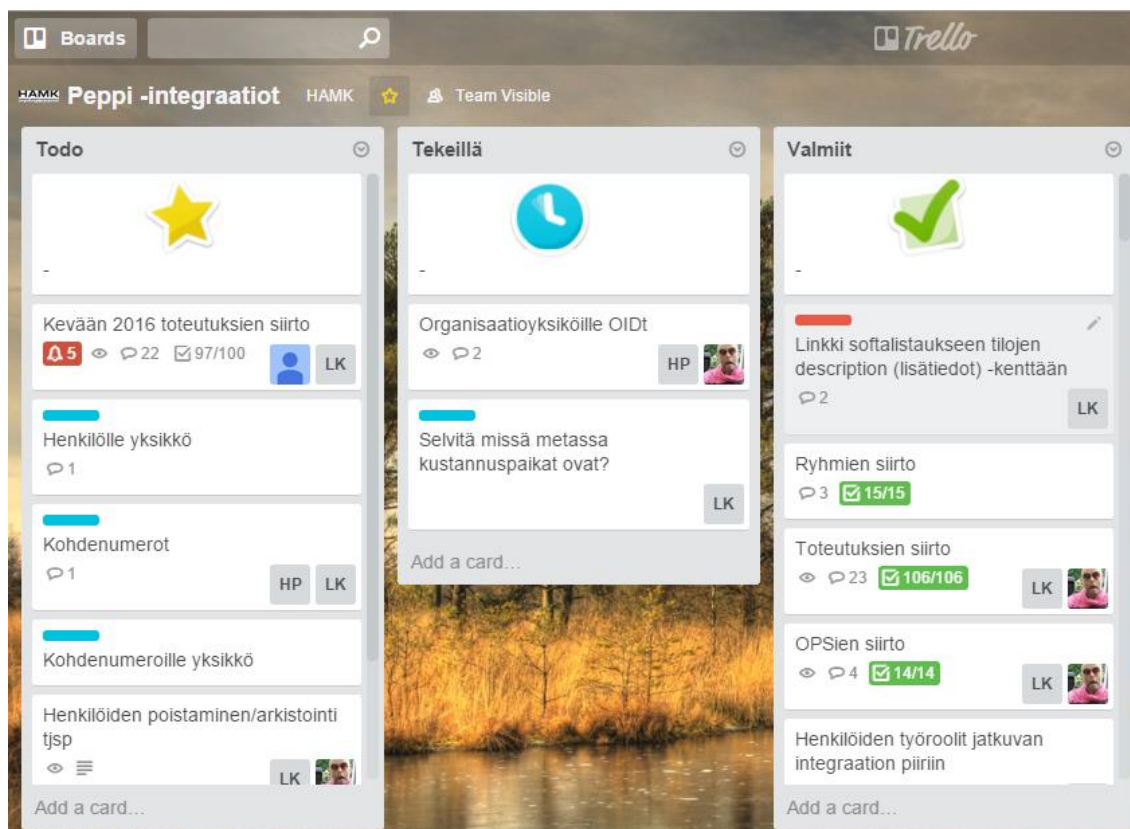
Sprintin päätyttyä kehitystiimi ja tuoteomistaja kokoontuvat suunnittelemaan seuraavan sprintin sisällön eli määrittelevät sen, mitä ominaisuuksia seuraavassa sprintissä toteutetaan.

HAMKin johto haluaa olla entistä paremmin selvillä siitä, miten eri projektit edistyvät. Projektien edistymisen ja mahdollisten ongelmakohtien tulee siis olla reaaliaikaisesti katseltavissa. Reaaliaikainen tarkasteltavuus tarkoittaa käytännössä sitä, että projektipäälliköt eivät voi mitenkään pitää manuaalisesti yllä ajantasaisia statusraportteja, vaan status-tiedon on synnyttävä automaattisesti projektin normaalin työskentelyn ohessa. Hyvä ratkaisu tähän olisi jonkin selainkäyttöisen projektinhallintapalvelun hyödyntäminen. Verkkopalveluiden uudistamisprojektissa käytetty Asana (<http://www.asana.com>) osoittautui projektin aikana toimivaksi vaihtoehdoksi, mutta se sai negatiivistakin palautetta lähinnä muilta kuin IT-ammattilaisilta. Toisaalta Asanaan on olemassa runsaasti erilaisia valmiita integraatioita, kuten Sprintboards-integraatio. Sprintboards on pilvipalvelu, joka kerää Asanan projektien tiedot ja koostaa niistä automaattisesti ajantasaisen yhteenvedon johdolle ja muille sidosryhmille (kuva 9).



KUVA 9. Sprintboard, jossa näkyy Asana-projektista automaattisesti haetut statustiedot (<http://www.sprintboards.com>)

Parhaillaan meneillään olevissa projekteissa olemme käyttäneet myös Trello-nimistä projektinhallinnan pilvipalvelua (<http://www.trello.com>). Trello on toiminnollisuuksiltaan selkeästi Asanaa suppeampi, mutta vastaavasti käyttäjiemme mielestä helppokäyttöisempi. Trello on rakennettu Kanban-menetelmän ympärille. Japaninkielinen sana Kanban tarkoittaa taulua. Taulu (kuva 10) on jaettu yksinkertaisimmillaan kolmeen listaan: Tehtävät, Tekeillä, Valmiit (Leido 2012, 177). Tehtävät-lista vastaa scrum-menetelmän backlogia ja Tekeillä-lista sprintin tehtävälistaa. Tehtävät-lista on järjestetty niin, että tärkeimmät tehtävät ovat listalla ensimmäisinä. Tiimiläiset poimivat Tehtävät-listalta toteutettavikseen tehtäviä ja siirtävät ne Tekeillä-listaan. Kun tehtävä on valmis, kehittäjä siirtää sen Valmiit-listaan. Jos Trelloa haluttaisiin käyttää vielä paremmin scrum-menetelmää tukevalla tavalla, niin Trelloon voitaisiin luoda neljäs lista nimeltään ”Sprintin tehtävät”. Sprintin suunnittelutilaisuudessa alkavaan sprinttiin valitut tehtävät siirrettäisiin sprintin Tehtävät-listalle, josta kehittäjät sitten poimisivat ne toteutettaviksi (liite 5).



KUVA 10. Trello-palvelun taulu ja listat.

5.5 Tiimin ja projektijohdon kehittyminen

Tiimin työskentelyn tehokkuuden ja projektin lopputuloksen kannalta on erittäin tärkeää, että tiimi suorittaa kunkin iteraation jälkeen itsearvioinnin. Itsearviointi tehdään yhdessä maksimissaan neljän tunnin mittaisessa tilaisuudessa, jota kutsutaan retrospektiiviksi. Tähän itsearviointiin totesimme verkkopalveluiden kehittämisprojektissa toimivaksi malliksi liitteen 2 mukaisen dokumenttipohjan. Vakioitun mallin käyttäminen on tärkeää, jotta voidaan tarvittaessa verrata projektin eri vaiheiden itsearviointien tuloksia toisiinsa. Vakioitu mallipohja myös tehostaa ajankäyttöä retrospektiiveissä, kun osallistujat voivat dokumentin muotoilun sijaan keskittyä olennaiseen.

Tarkastellessani jälkikäteen verkkopalveluiden kehittämisprojektin retrospektiivejä, havaitsin, että hyvin usein projektia koskevat kehittämis- tai muutosehdotukset liittyivät siihen, miten johdon suuntaan saataisiin viestittyä projektin tärkeitä asioita ja mahdollisia ongelmatilanteita. Tällaiseen jatkuvaan projektin johdon suuntaan tapahtuvaan palautteenantoon voisi olla hyvä käyttää vakioituja lomakkeita. Kuvassa 11 näkyy esimerkki

palautelomakkeesta, jonka avulla tiimin jäsenet voivat viestiä projektipäällikölle projektin tavoitteiden saavuttamista uhkaavia tekijöitä.

Improvement Initiative	Score (1-3)
Time wasted in meetings	
Delayed achievement of important results	
Outdated project's website	
Time wasted in avoidable errors	
Time wasted in irrelevant tasks	
Poor communications	
Others	

KUVA 11. Palautelomake tiimiltä projektipäällikölle (Leido 2012, 141),

Pisteytys toimii niin, että kukin palautteen antaja antaa häiritseville pisteitä niin, että helpoimmin poistettaville häiritseville annetaan kolme pistettä ja vaikeimmin poistettaville häiritseville yksi piste (Leido 2012, 142). Taulukon täyttäminen ei oletettavasti vie paljoa aikaa ja se antaa projektipäällikölle selkeän, priorisoidun listan tarvittavista korjaustoimenpiteistä. Ensimmäiseksi korjataan eniten pisteitä saaneet asiat, koska ne ovat helpoimmat korjata ja näin saadaan tiimille mahdollisimman nopeaksi nähtäville se, että projektin johto reagoi palautteisiin. Liitteessä 1 esitetään HAMKin käyttöä varten sovellettu versio tästä palautelomakkeesta, kyseessä olevan lomakkeen väittämät ovat esimerkinomaisia.

Ohjausryhmänkin suuntaan annettavaa palautetta olisi hyvä kerätä ja tarkoitukseen voitaisiin käyttää samankaltaista taulukkoa (Kuva 12). Anonyymisti kerättävä palaute mataltaa palautteenantokynnystä ja toisaalta valmiin taulukon avulla varmistetaan se, että palaute pysyy asiassa ja asiallisena. Tämä palautelomake käyttää samanlaista pisteytystapaa kuin edellä esiteltykin lomake. Ohjausryhmään kohdistuvan palautteen kerääminen tapahtuisi kunkin iteraation lopuksi ja palautteen antajina olisi kehitystiimi, projektipäällikkö ja tuoteomistaja.

Improvement Initiative	Score (1-3)
Delay on approvals/decisions	
Poor response to "signals"	
Too much time on status reports	
Lack of resources availability	
Sudden changes to the project charter	
Disturbing micro-management	
Poor communication/Other waste or <i>muda</i>	

KUVA 12. Palautelomake tiimiltä ohjausryhmälle (Leido 2012, 142).

Liitteessä 5 esitetään HAMKin käyttöön mukailtu versio tästä palautelomakkeesta. Lomakkeen väittämät ovat esimerkinomaisia.

6 POHDINTA

Nopeasti muuttuva toimintaympäristö ohjaa HAMKia muuttamaan kehittämisen ja projektinhallinnan menetelmiään ja organisaatiollamme onkin suuret odotukset uusien menetelmien suhteen.

Kuten verkkopalveluiden uudistamisprojekti osoitti, ketterät projektinhallintamenetelmät ja uudenlaiset hankintamallit voivat parhaimmillaan tarjota organisaatiollemme keinoja entistä parempien palveluiden kehittämiseen entistä kustannustehokkaammin. Ketterien menetelmien käyttö tuo asiakkaille hyötyjä nopeammin ja tehty työ näkyy sidosryhmille välittömästi.

On kuitenkin hyvä muistaa, että ketterät menetelmät eivät ole mikään lisäarvautomaatti, vaan niiden käyttöönotto vaatii pitkäjännitteisyyttä ja organisaation on oltava valmis tekemään suuriakin muutoksia toimintatapoihinsa kuten henkilöiden toimenkuviin, töiden organisointiin ja hankintatapoihin.

Ketterien menetelmien käyttöönotto jatkuu HAMKissa vuoden 2015 aikana niin, että tämän opinnäytetyön kirjoittaja laatii organisaatiollemme ketterien projektien menetelmäoppaan, joka tarjoaa projektipäälliköille toimintaohjeita ja dokumenttipohjia ketterien projektien ohjaamiseen. Tässä opinnäytetyössä ja siihen liittyvässä kehittämis-tehtävässä keskityttiin lähes yksinomaan scrum-menetelmään, mutta organisaatiollemme sopivin ketterän projektinhallinnan malli voisi olla esimerkiksi scrum- ja lean-menetelmien sekoitus. Tällaisen yhdistetyn menetelmän tutkiminen ja pilotointi olisikin ehdottomasti tarpeen.

Ours is too great and too complex a nation for even such as I to direct and lead every action. (Attila The Hun).

Mahtavinkaan meistä ei pysty hallinnoimaan kaikkia asioita. Siksi minulle projektipäällikkönä onkin ollut suuri ilo ja helpotus tutustua ja ottaa käyttöön ketterät projektinhallinnan menetelmät, koska ne tuovat yksinkertaisuutta yhä monimutkaistuviin projekteihin ja mahdollistavat tiimin ja sen yksilöiden kehittymisen.

LÄHTEET

The Agile School, 2012. Types of Scrum Master; Do we need someone with IT background as a Scrum Master?, verkkosivu. Luettu 27.9.2015

<http://www.theagileschool.com/2012/02/types-of-scrum-master-do-we-need.html>

CC Pace Systems, 2003. Agile Project Management. Ladattu 5.3.2015.

http://www.ccpace.com/asset_files/AgileProjectManagement.pdf

Chin, G. (2003). Agile Project Management: How to Succeed in the Face of Changing Requirements. New York, USA: Amacom.

Helsingin Yliopisto/Humanistinen tiedekunta, 2006. Mikä on projekti?, verkko-opinimateriaali. Luettu 6.6.2015.

<http://www.ling.helsinki.fi/kit/2006k/clt310pro/yleista/maaritelma.shtml>

Highsmith, J. 2002. Agile Software Development Ecosystems. Westford: Addison-Wesley

InformIT, 2006. Perils and Pitfalls of Agile Adoption, verkkosivu. Luettu 1.11.2015.

<http://www.informit.com/articles/article.aspx?p=441115>

Kivekäs, O. 2013. Konsulttikratia, blogikirjoitus. Luettu 26.5.2015.

<http://otsokivekas.fi/2013/07/konsulttikratia/>

Koivisto, M. 2012. Onnistumisia minikilpailulla, blogikirjoitus. Luettu 19.3.2015.

<http://www.ptcs.fi/fi/blogi/puite+minikilpailu+julkiset+hankinnat>

Laki julkisista hankinnoista 30.3.2007/348.

Larman, C. 2004. Agile & iterative development – a manager's guide. Westford: Addison-Wesley

Layton, M. 2012. Agile project management for dummies. Hoboken: John Wiley & Sons.

Leido, P. 2014. Lean & Agile Project Management. Bloomington: Trafford Publishing

Lekman Consulting, 2014. Suomenkielinen Scrum Guide, pdf -dokumentti. Ladattu 9.9.2015

<https://scrumwell.files.wordpress.com/2014/03/scrum-guide-2013-fi-v1-1.pdf>

Moore, Geoffrey A. 2006. Crossing the Chasm. Marketing and Selling Disruptive Products to Mainstream Customers. Revised edition. New York: HarperCollins Publishers

Notarangelo, J. 2010. Jack Notarangelo's Application Engineering Blog - Humphrey's Requirements Uncertainty Principle. Luettu 17.3.2015
<http://applicationengineering.blogspot.fi/2010/11/humphreys-requirements-uncertainty.html>

Openview Labs, 2012. 3 Big Benefits of Scrum and Why It Beats Waterfall for Product Development, verkkosivu. Luettu 2.11.2015.
<http://labs.openviewpartners.com/3-benefits-of-scrum-beats-waterfall/>

Poimala, S. 2015. Ketteryys haltuun: Scrum pähkinänkuoressa, verkkosivu. Luettu 5.5.2015. <https://www.meteoriitti.com/Artikkelisarjat/Ketteryys-haltuun/Ketteryys-haltuun-Scrum-pahkinankuoressa/>

Project Management Institute, 2015. What is Project Management?, verkkosivu. Luettu 9.6.2015
<http://www.pmi.org/About-Us/About-Us-What-is-Project-Management.aspx>

Saaranen-Kauppinen, A. & Puusniekka, A. 2006. KvaliMOTV - Menetelmäopetuksen tietovaranto, luku 6.3.2 Teemahaastattelu, verkkosivu. Luettu 1.8.2014
http://www.fsd.uta.fi/menetelmaopetus/kvali/L6_3_2.html

Schwaber, K. 2004. Agile project management with scrum. Redmond: Microsoft Press.

Scrum Alliance, 2009. The Product Vision, verkkosivu. Luettu 22.6.2015
<https://www.scrumalliance.org/community/articles/2009/january/the-product-vision>

Scrum Alliance, 2012. The Value of Release Planning, verkkosivu. Luettu 27.9.2015
<https://www.scrumalliance.org/community/articles/2012/august/the-value-of-release-planning>

Scrum Alliance, 2015. Hybrid Agile Versus Agile, verkkosivu. Luettu 2.11.2015.
<https://www.scrumalliance.org/community/articles/2015/may/hybrid-agile-versus-agile>

Techrepublic. 2009. The roots of agile project management. Luettu 6.2.2015.
<http://www.techrepublic.com/blog/tech-decision-maker/the-roots-of-agile-project-management/>

TEM Työ- ja elinkeinoministeriö. 2015. Kynnysarvot, verkkosivu. Luettu 29.4.2015.
<http://www.hankintailmoitukset.fi/fi/docs/kynnysarvot>

Toivanen, E. 2014. Ketteryys tietojärjestelmän hankinnassa ja toteutuksessa, esitys korkeakoulujen IT-päivillä 4.11.2014. Nähty 4.11.2014 ja ladattu 11.4.2015.

Udemy, 2013. Agile Vs. Waterfall: Evaluating The Pros And Cons, verkkosivu. Luettu 28.9.2015
<https://blog.udemy.com/agile-vs-waterfall/>

Vizdos, M. 2006. The Classic Story of the Pig and Chicken, verkkosivu Luettu 21.5.2015.

<http://www.implementingscrum.com/2006/09/11/the-classic-story-of-the-pig-and-chicken/>

Vänskä, O. 2015. Ketterän koodin suurkaupunki. TIVI 10/2015, 34-38.

The Wall Street Journal. 2009. What is the Difference Between Management and Leadership?, verkkosivu. Luettu 24.5.2015.

<http://guides.wsj.com/management/developing-a-leadership-style/what-is-the-difference-between-management-and-leadership/>

LIITTEET

Liite 1. Puitekilpailutuksen ehdottomat referenssivaatimukset

vähintään yksi Sharepoint on premise -käyttöönoton konsultointi mukaan lukien infrastruktuurin suunnittelu, asennus ja ylläpito

vähintään yksi toteutus Sharepointiin kehitetyistä räätälöidyistä sovelluksista

vähintään yksi toteutus Sharepointin käyttöliittymäsuunnittelusta ja -toteutuksesta räätälöidyillä mallipohjilla

vähintään yksi Sharepoint Online -käyttöönoton konsultointi

vähintään yksi hybridimallin (Sharepoint on premise + Sharepoint Online) mukainen Sharepoint-käyttöönoton konsultointi

vähintään yksi toteutus Sharepoint on premisen ja Sharepoint Onlinen välisistä integraatioista

vähintään yksi toteutus Sharepoint-informaatioarkkitehtuurin suunnittelusta ja toteutuksesta

Liite 2. Sprinttien retrospektiivi –lomake

Sharepoint-projektin 10. sprintin retrospektiivi

Edellisten muutosten toteutuminen

Tiimin tunnelmat

Mikä meni hyvin?

Mikä meni vähemmän hyvin?

Mitä konkreettista muutamme tiimimme tai projektin toiminnassa, jotta onnistumme paremmin, ja miksi?

Liite 3. Palautealomakeluonnos tiimiltä projektipäällikölle

**Hämeen ammattikorkeakoulu
Oy**

Projektin nimi

Kehitystiimin palaute projektipäällikölle

Kehitysehdotus	Pisteet (1-3)
Kokouksiin menee liian paljon aikaa	
Tuloksia saavutetaan liian hitaasti	
Aikaa hukataan turhiin virheisiin	
Aikaa hukataan turhiin tehtäviin	
Viestintä ei toimi	
Muut	

(anna kolme pistettä mielestäsi helpoimmin poistettavalle)

Liite 4. Palautelomakeluonnos kehitystiimiltä ohjausryhmälle

**Hämeen ammattikorkeakoulu
Oy**

Projektin nimi

Kehitystiimin palaute ohjausryhmälle

Kehitysehdotus	Pisteet (1-3)
Päätöksenteko/hyväksyntä kestää liian kauan	
Signaaleihin reagoidaan epämääräisesti	
Projektin statusraportointiin kuluu liikaa aikaa	
Projektin resursseja ei saada turvattua	
Projektin visio muuttuu	
Mikromanagerointi	
Viestintä ei toimi	
Mudaa	

(anna kolme pistettä mielestäsi helpoimmin poistettavalle)

Liite 5. Scrum –menetelmää varten mukautettu Trello-taulu

